

UNIVERSIDADE DE CAXIAS DO SUL
CENTRO DE COMPUTAÇÃO E TECNOLOGIA DA INFORMAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

SAMUEL BRANDO OLDRA

**Integração de Dados Biológicos
- Proteínas e Doenças Gênicas -**

Prof. MSc. Daniel Luís Notari
Orientador

Caxias do Sul, Dezembro de 2009

*“Não temeis a grandeza;
alguns nascem grandes,
alguns conseguem grandeza,
a alguns a grandeza lhes é imposta
e a outros a grandeza lhes fica grande.”*

— WILLIAM SHAKESPEARE

AGRADECIMENTOS

Primeiramente gostaria de agradecer aos meus pais por toda a ajuda que me deram durante a minha vida e por acreditarem em mim, mesmo quando eu não sabia se era capaz.

Gostaria de agradecer a minha família e aos meus amigos pela compreensão de que muitas vezes não pude estar presente, por estar buscando algo que me tornasse uma pessoa melhor.

Também gostaria de agradecer aos meus colegas que sempre se mostraram companheiros na hora de se reunir para fazer um trabalho de faculdade ou, então, ir para um barzinho tomar cerveja e dar risada da vida.

Por fim, gostaria de agradecer aos professores por sempre estarem dispostos a passar seus conhecimentos, esclarecer dúvidas, ajudar na solução de problemas e participarem de pesquisas.

Na minha vida sempre tive bem claro que não conseguiria mudar o mundo sozinho, mas que nada me impediria de tentar. Hoje sei que sou capaz de mudar o mundo, isso porque sei que não estou sozinho nessa jornada. A todas as pessoas que de alguma forma fazem parte da minha história gostaria de dizer, obrigado!

RESUMO

Atualmente existem vários repositórios de dados biológicos espalhados pela Internet e podem ser encontradas informações dos mais diferentes interesses através da utilização combinada de alguns deles. Para tanto, já foram documentados diversos processos biológicos para obtenção de determinados resultados, e em sua grande maioria eles são executados de forma manual, o que aumenta a chance de erros. O objetivo desse trabalho é modelar e desenvolver um sistema capaz de pesquisar doenças genéticas, automatizando e integrando os *sites* e o software utilizados pelo especialista. Esse trabalho também apresenta um estudo sobre os temas relacionados ao processo de pesquisa de uma doença genética e outros repositórios de dados biológicos analisando a informação que eles disponibilizam.

Palavras-chave: Biologia Molecular, Processo Biológico, Dados Biológicos, Doenças Genéticas, Workflow Científico, Integração de Dados, Modelagem de Sistemas.

Integration of Biological Data - Proteins and Genetic Diseases

ABSTRACT

Currently there are several repositories of biological data around the Internet and we can find information from different interests through the combined use of some of them. To that end, have been documented several biological processes to achieve certain results, and for the most part they are run manually, which increases the chance of errors. The aim of this paper is to model and develop a system capable of searching for genetic diseases by automating and integrating the sites and software used by the specialist. This work also presents a study on issues related to the research process of a genetic disease and repositories of biological data, by analyzing the information they provide.

Keywords: Molecular Biology, Biological Process, Data Biological, Genetic Diseases, Scientific Workflow, Data Integration, Modeling Systems.

LISTA DE FIGURAS

2.1	Aminoácidos e seus radicais	17
2.2	Esquema de formação de um dipeptídeo	18
2.3	Estrutura da proteína	22
2.4	Duplicação, transcrição e tradução	23
2.5	Nucleotídeos do DNA	24
2.6	Esquema de molécula de DNA	25
2.7	Esquema de molécula de RNA	25
2.8	Esquema de duplicação de DNA	26
2.9	Duplicação do DNA	27
2.10	Esquema de transcrição de RNA	27
2.11	Transcrição de DNA em RNA	28
2.12	Correspondência entre DNA, RNA e Aminoácidos	30
2.13	Síntese protéica	31
3.1	Rede de livre-escala	38
3.2	Fluxo de pesquisa	40
3.3	Pesquisa da doença	41
3.4	Lista de ocorrências da doença	41
3.5	Relatório da doença	42
3.6	Localizando proteínas/genes no relatório	42
3.7	Pesquisa da proteína	42
3.8	Lista de organismos que possuem a proteína	43
3.9	Apresentação da rede de interação da proteína	43
3.10	Seleção do tipo de arquivo da rede de interação	43
3.11	Importando rede de interação da(s) proteína(s)	44
3.12	Merge das redes de interações das proteínas	44
3.13	Representação gráfica das redes de interações	45
3.14	Pesquisa das proteínas	46
3.15	Lista de organismos que possuem as proteínas	47
3.16	Lista de ocorrências das proteínas	48

3.17	Apresentação da rede de interação das proteínas	48
3.18	Seleção do tipo de arquivo das redes de interações	49
3.19	Representação gráfica das redes de interações	49
4.1	Caso de Uso	56
4.2	Fluxo de pesquisa do software	56
4.3	Diagrama de arquitetura da aplicação	58
4.4	Algoritmo de extração dos dados	59
5.1	Diagrama de arquitetura da aplicação	62
5.2	Diagrama de Componentes	62
5.3	Fluxo Software	71
5.4	Procura da doença	71
5.5	Seleciona a ocorrência da doença	72
5.6	Apresenta relatório da doença e seleciona proteínas	73
5.7	Seleciona as ocorrências das proteínas	74
5.8	Apresenta rede de interação da proteína e fornece arquivo XML .	75
5.9	Fluxo de pesquisa	76
6.1	Primeiro estudo de caso rede STRING	79
6.2	Primeiro estudo de caso rede Cytoscape	80
6.3	Segundo estudo de caso rede STRING	81
6.4	Terceiro estudo de caso rede STRING	83
6.5	Quarto estudo de caso rede STRING primeira pesquisa	86
6.6	Quarto estudo de caso rede STRING segunda pesquisa	88
6.7	Quarto estudo de caso rede Cytoscape segunda pesquisa	88
6.8	Quarto estudo de caso rede STRING terceira pesquisa	91
6.9	Quarto estudo de caso rede Cytoscape terceira pesquisa	92
7.1	Formulário de pesquisa (1/3)	119
7.2	Formulário de pesquisa (2/3)	120
7.3	Formulário de pesquisa (3/3)	121

LISTA DE TABELAS

2.1	Aminoácidos essenciais (1/2)	18
2.2	Aminoácidos essenciais (2/2)	19
2.3	Aminoácidos dispensáveis (1/2)	20
2.4	Aminoácidos dispensáveis (2/2)	21
2.5	Aminoácidos condicionalmente indispensáveis	21
2.6	Código genético	30
4.1	Regras de Grafia	55

LISTA DE SCRIPTS

5.1	step02.php (trecho)	63
5.2	step03.php (trecho)	64
5.3	step04.php (trecho)	65
5.4	step05.php (trecho)	66
5.5	flow.php (trecho)	68
5.6	createXMLFlow.php (trecho)	70
7.1	index.php	97
7.2	step01.php	97
7.3	step02.php	100
7.4	step03.php	103
7.5	step04.php	106
7.6	step05.php	108
7.7	flow.php	112
7.8	createXMLFlow.php	117

LISTA DE ABREVIATURAS E SIGLAS

CIB-DDBJ	<i>Center for Information Biology and DNA Data Bank of Japan</i>
DDBJ	<i>DNA Data Bank of Japan</i>
DNA	Ácido Desoxirribonucléico (<i>Deoxyribonucleic Acid</i>)
EBI	<i>European Bioinformatics Institute</i>
EMBL	<i>European Molecular Biology Laboratory</i>
HGNC	<i>HUGO Gene Nomenclature Committee</i>
HTML	<i>HyperText Markup Language</i>
MIM	<i>Mendelian Inheritance in Man</i>
NCBI	<i>National Center for Biotechnology Information</i>
NIG	<i>National Institute of Genetics</i>
OMIM	<i>Online Mendelian Inheritance in Man</i>
PDB	<i>Protein Data Bank</i>
PHP	<i>PHP: Hypertext Preprocessor</i>
RCSB	<i>Research Collaboratory for Structural Bioinformatics</i>
RNA	Ácido Ribonucléico (<i>Ribonucleic Acid</i>)
SGBD	Sistema Gerenciador de Banco de Dados
SIB	<i>Swiss Institute of Bioinformatics</i>
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1 INTRODUÇÃO	14
2 BIOINFORMÁTICA	16
2.1 Proteínas	16
2.1.1 Aminoácidos: as unidades de construção da proteína	17
2.1.2 A estrutura da proteína	19
2.1.3 Proteínas: relação entre forma e função	22
2.2 O dogma central da biologia molecular	23
2.2.1 A estrutura do DNA e do RNA	23
2.2.2 Replicação de DNA	24
2.2.3 Transcrição de DNA	26
2.2.4 Tradução de mRNA	29
2.2.5 Síntese de proteínas	31
2.2.6 Genomas e genes	32
2.2.7 Evolução molecular	32
2.3 Bancos de dados biológicos	32
2.3.1 DDBJ	33
2.3.2 EMBL-EBI	33
2.3.3 GenBank	34
2.3.4 OMIM	34
2.3.5 PDB	34
2.3.6 STRING	35
2.3.7 Swiss-Prot	35
2.4 Considerações finais	35
3 BIOLOGIA DE SISTEMAS	36
3.1 Redes de livre-escala	36
3.1.1 Redes de sinalização	37
3.1.2 Redes celulares	38

3.1.3	Redes metabólicas	38
3.1.4	Redes de interação de proteínas	39
3.2	Ontologia gênica	39
3.3	Fluxo de pesquisa de uma doença	40
3.3.1	Descrição do Fluxo A	40
3.3.2	Descrição do Fluxo B	46
3.4	Considerações finais	49
4	PROPOSTA DE SOFTWARE	51
4.1	Workflow	51
4.1.1	VisTrails	51
4.1.2	Kepler	52
4.1.3	Taverna	53
4.1.4	Egene	53
4.2	Nomenclatura genética	53
4.3	Modelagem do software	55
4.3.1	Modelo de negócio - caso de uso	55
4.3.2	Workflow científico para análise de redes de interação protéica	56
4.3.3	Diagrama de arquitetura da aplicação	57
4.4	Detalhamento da implementação	58
4.4.1	Algoritmo de extração dos dados	58
4.4.2	Acesso aos sites do OMIM e do STRING	59
4.4.3	Salvar e recuperar informações	59
4.4.4	Interação com o software Cytoscape	59
4.5	Considerações finais	60
5	IMPLEMENTAÇÃO	61
5.1	Diagrama de arquitetura da aplicação	61
5.2	Implementação do sistema web	61
5.2.1	Pesquisa da doença	63
5.2.2	Busca e seleção da doença	63
5.2.3	Visualização do relatório e seleção das proteínas	64
5.2.4	Seleção das ocorrências das proteínas	65
5.2.5	Visualização da rede de interação da(s) proteína(s)	66
5.2.6	Documentação do fluxo de pesquisa da doença	67
5.2.7	Salvamento do fluxo de pesquisa da doença	69
5.2.8	Integração com o software Cytoscape	69
5.3	Workflow científico do sistema web	70
5.4	Considerações finais	77

6 ESTUDO DE CASO	78
6.1 Primeiro estudo de caso	78
6.2 Segundo estudo de caso	79
6.3 Terceiro estudo de caso	81
6.4 Quarto estudo de caso	83
6.4.1 Primeira pesquisa	83
6.4.2 Segunda pesquisa	86
6.4.3 Terceira pesquisa	89
6.5 Considerações finais	91
7 CONCLUSÃO	93
REFERÊNCIAS	94
ANEXOS	97

1 INTRODUÇÃO

A bioinformática é uma ciência aplicada que surgiu do casamento da biologia e da informática. Hoje a computação contribui para a biologia com a oferta de desde processamento bruto e armazenamento de dados biológicos, até sofisticados métodos matemáticos necessários para alcançar resultados (LESK, 2008).

Após o Projeto Genoma Humano ter completado sua fase inicial, cientistas e políticos começaram a articular cada vez mais, visões de como a tecnologia orientada para a aquisição de conhecimento genômico poderia ser transformada em estratégias de intervenção. A área em que muitas destas ambições e esperanças convergiram é agora o que é chamado de biologia de sistemas. O objetivo global da biologia de sistemas é o objetivo final da biologia moderna, a obtenção de uma fundamental, abrangente e sistemática compreensão da vida (O'MALLEY; DUPRE, 2005).

Atualmente existem inúmeras bases de dados com informações biológicas espalhadas por diversos centros de pesquisa, normalmente esses dados pouco informam quando vistos de forma isolada. Para ajudar nessa etapa a informática passou a adquirir importância dentro da área da biologia, sua função é tentar transformar esses dados em informações úteis (GIBAS; JAMBECK, 2001; LESK, 2008).

O grande problema está em extrair o significado destes dados, pois cada centro desenvolveu a sua forma de organizar as informações, muitas vezes contendo a mesma informação representada de forma diferente. Devido a essas bases de dados terem sido criadas sem um padrão universal de acesso aos dados, o processo de compartilhamento destes dados entre os pesquisadores se torna difícil e trabalhoso (GIBAS; JAMBECK, 2001).

Os cientistas dessa área normalmente realizam seus experimentos sem registrar a seqüência de atividades realizadas. Isto dificulta a possibilidade de repetir o experimento. Para isto, é necessário registrar as atividades feitas com os softwares usados, parâmetros utilizados, etc. Isto pode ser feito com o uso de *workflow*.

Workflows científicos são *workflows* voltados para modelagem e resolução de problemas científicos através de técnicas tradicionais de *workflows*, ou seja, as idéias de execução de um conjunto de tarefas em uma determinada seqüência, foram aproveitadas na área científica para a realização de experimentos e estudos (DI-GIAMPIETRI, 2007; SILVA, 2006).

O problema que será abordado nesse trabalho será integrar os dados de proteínas com os de doenças ligadas às mesmas, atualmente esse processo é feito manualmente, através de um *workflow* executado de forma manual e demanda trabalho dos profissionais de biologia.

Doenças causadas por alterações nos genes são geralmente conhecidas como doenças gênicas (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

Durante o desenvolvimento da solução serão analisadas formas de integrar esses dados, usando a ontologia gênica e disponibilizá-los para validação e exportação para as devidas ferramentas, o que tornará o trabalho do especialista mais produtivo e menos repetitivo.

No segundo capítulo, Bioinformática, são apresentados conceitos gerais da biologia molecular e fontes de dados biológicos. No terceiro capítulo, Biologia de sistemas, são apresentados conceitos gerais da biologia de sistemas, e é explicado o que é a ontologia gênica e também como é o fluxo de pesquisa de uma doença gênica, estudado nesse trabalho. No quarto capítulo, Proposta de software, são apresentados os conceitos e algumas ferramentas de *workflow* científico, é explicado o que é e como funciona a nomenclatura genética e são apresentados os artefatos da proposta de software. No quinto capítulo, Implementação, são apresentados os artefatos e *scripts* do sistema, e também um manual de uso do software. E, por fim, no sexto capítulo, Estudo de caso, são apresentados os estudos de caso realizados com alguns profissionais da área da biologia e da informática.

2 BIOINFORMÁTICA

A bioinformática é uma ciência aplicada que surgiu do casamento da biologia e da informática, e hoje é denominada como a ciência de usar as informações para entender a biologia. Também se pode dizer que a bioinformática é um subconjunto de um campo maior da biologia computacional, que visa à aplicação de técnicas analíticas quantitativas à modelagem de sistemas biológicos (GIBAS; JAMBECK, 2001; LESK, 2008).

A pesquisa em bioinformática e biologia computacional pode compreender desde a abstração das propriedades de um sistema biológico em um modelo matemático ou físico até a implementação de novos algoritmos para a análise de dados, o desenvolvimento de bancos de dados e das ferramentas *web* para acessá-los (GIBAS; JAMBECK, 2001; LESK, 2008).

Nas seções que seguem será explicado o papel das proteínas e como elas funcionam, o dogma central da biologia molecular, ou seja, o processo pelo qual o DNA se replica ou é transcrito em RNA, e por sua vez o RNA é traduzido em proteínas, e serão apresentados brevemente os principais bancos de dados biológicos.

2.1 Proteínas

O papel das proteínas e dos ácidos nucléicos está diretamente relacionado ao controle de tudo o que a célula é e o que ela faz. As proteínas são componentes obrigatórios dos seres vivos, aparecendo até nos vírus, que não tem estrutura celular. As proteínas desempenham três papéis fundamentais: a construção da matéria viva, reposição de material celular desgastado e crescimento são funções que dependem da fabricação de proteínas pelos ribossomos da célula; a regulação do metabolismo celular, função desempenhada pelas enzimas e sem as quais as reações químicas numa célula não seriam possíveis; e defesa do organismo das invasões de agentes externos, desempenhada pelos anticorpos (S. JUNIOR; SASSON, 2003).

Nas subseções que seguem será explicado o que é e como funcionam os aminoácidos, as unidades de construção da proteína, a estrutura da proteína e a relação entre forma e função nas proteínas.

2.1.1 Aminoácidos: as unidades de construção da proteína

As proteínas são moléculas grandes e de estrutura complexa. Uma molécula de proteína é constituída por muitas unidades menores, ligadas entre si, os aminoácidos. Qualquer molécula de aminoácido tem um grupo ácido carboxílico ($-COOH$) e um grupo amina ($-NH_2$) ligados a um átomo de carbono. A esse carbono ficam ligados um átomo de hidrogênio e um radical (R). Como pode ser visto na Figura 2.1 o radical pode ser um simples átomo de hidrogênio (na glicina), um $-CH_3$ (na alanina), ou grupos mais complexos como nos outros. Os vinte aminoácidos existentes na natureza diferem entre si apenas quanto ao seu radical (BERG; TYMOCZKO; STRYER, 2004; S. JUNIOR; SASSON, 2003).

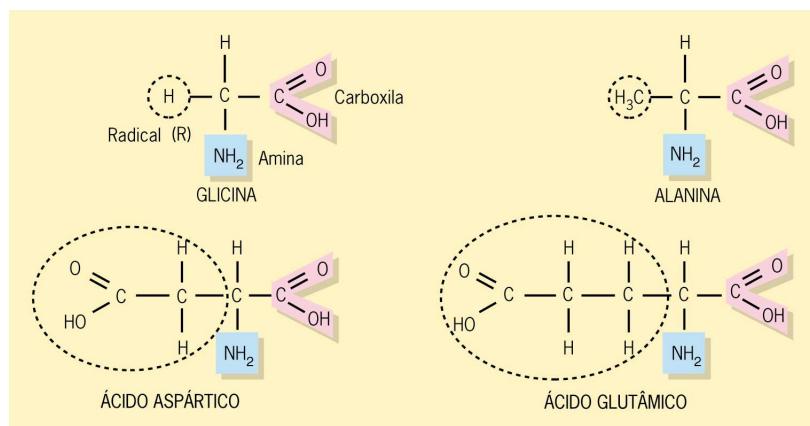


Figura 2.1: Aminoácidos e seus radicais
Fonte: (S. JUNIOR; SASSON, 2003)

Pode-se dividir os aminoácidos em dois tipos: os naturais e os essenciais. Os aminoácidos naturais são os que um organismo animal é capaz de produzir. Os aminoácidos essenciais são aqueles que os animais precisam ingerir, já que são obrigatórios para a síntese de suas proteínas e para sua sobrevivência. Os vegetais são capazes de produzir os vinte aminoácidos necessários conhecidos à produção de suas proteínas (S. JUNIOR; SASSON, 2003).

Dois aminoácidos se unem na molécula de proteína através de uma ligação peptídica. Como ilustra a Figura 2.2 a reação ocorre entre a carboxila de um aminoácido e a amina de outro, havendo perda de uma molécula de água, trata-se de uma síntese por desidratação. O produto formado quando dois aminoácidos se ligam é chamado de dipeptídeo, o tripeptídeo e o tetrapeptídeo são formados, respectivamente, por três e quatro aminoácidos (NELSON; COX, 2006).

Quando ocorre um número maior de aminoácidos na molécula, usa-se o termo polipeptídeo, o termo proteína é usado para designar peptídeos com número superior a setenta aminoácidos (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

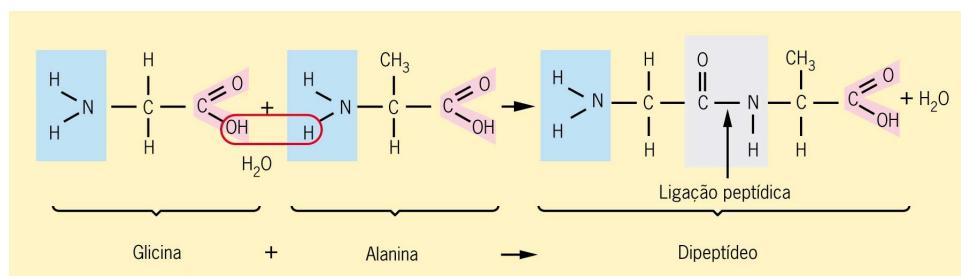


Figura 2.2: Esquema de formação de um dipeptídeo
Fonte: (S. JUNIOR; SASSON, 2003)

Os aminoácidos são blocos formadores de proteínas e tecido muscular. Todos os tipos de processo fisiológico como energia, recuperação, ganhos de músculos, força e perda de gordura, assim como funções do cérebro e temperamento, estão inteiramente ligados aos aminoácidos. Eles também podem ser convertidos e enviados diretamente para o ciclo de produção de energia do músculo (MOTTA, 2005; NELSON; COX, 2006).

São 20 os aminoácidos construtores moleculares de proteínas. De acordo com uma classificação aceita, nove são chamados de aminoácidos essenciais, significando que são fornecidos por algum alimento ou fonte de suprimento. E os demais, chamados aminoácidos dispensáveis ou condicionalmente indispensáveis, são baseado na habilidade do organismo em sintetizá-los de outros aminoácidos (CHAMPE; HARVEY; FERRIER, 2006). Nas Tabelas 2.1 e 2.2 são apresentados os aminoácidos essenciais, com a sua abreviatura e função.

Aminoácidos Essenciais		
Abrev.	Aminoácido	Função
His	Histidina <i>Histidine</i>	absorve ultravioleta na pele. É importante na produção de células vermelhas e brancas, sendo usado no tratamento de anemias, doenças alérgicas, artrite, reumatismo e úlceras digestivas;
Ile	Isoleucina <i>Isoleucine</i>	essencial na formação de hemoglobina. É usado para a obtenção de energia pelo tecido muscular e para prevenir perda muscular em pessoas debilitadas;
Leu	Leucina <i>Leucine</i>	usado como fonte de energia, ajuda a reduzir a queda de proteína muscular. Modula o aumento dos precursores neurotransmissores pelo cérebro, assim como a liberação das encefalinas, que impedem a passagem dos sinais de dor para o sistema nervoso. Promove cicatrização da pele e de ossos quebrados;

Tabela 2.1: Aminoácidos essenciais (1/2)

Aminoácidos Essenciais		
Abrev.	Aminoácido	Função
Lys Lis	Lisina <i>Lysine</i>	inibe vírus e é usado no tratamento de herpes simples. Ajuda no crescimento ósseo, auxiliando a formação do colágeno, a fibra protéica que produz ossos, cartilagem e outros tecidos conectivos. Baixos níveis de lisina podem diminuir a síntese protéica, afetando os músculos e tecidos de conexão. Este aminoácido, combinado à vitamina C, forma a l-carnitina, um bioquímico que possibilita ao tecido muscular usar oxigênio com mais eficiência, retardando a fadiga;
Met	Metionina <i>Methionine</i>	precursor da cistina e da creatina, ajuda a aumentar os níveis antioxidantes (glutathione) e reduzir os níveis de colesterol no sangue. Também ajuda na remoção de restos tóxicos do fígado e na regeneração deste órgão e dos rins;
Phe Fen	Fenilalanina <i>Phenylalanine</i>	maior precursor da tirosina, melhora o aprendizado, a memória, o temperamento e o alerta mental. É usado no tratamento de alguns tipos de depressão. Elemento principal na produção de colágeno, também tira o apetite;
Thr The	Treonina <i>Threonine</i>	desintoxicante, ajuda a prevenir o aumento de gordura no fígado. Componente importante do colágeno, é encontrado em baixos níveis nos vegetarianos;
Trp Tri	Triptofano <i>Tryptophan</i>	é utilizado pelo cérebro na produção de serotonina, um neurotransmissor que leva as mensagens entre o cérebro e um dos mecanismos bioquímicos do sono existentes no organismo, portanto oferecendo efeito calmante. Encontrado nas fontes de comidas naturais, promove sonolência, por isso deve ser consumido à noite;
Val	Valina <i>Valine</i>	não é processado pelo fígado, mas é ativamente absorvido pelos músculos, sendo fundamental no metabolismo dos ácidos líquidos adiposos. Influencia a tomada, pelo cérebro, de outros neurotransmissores (triptofano, fenilalanina, tirosina).

Tabela 2.2: Aminoácidos essenciais (2/2)

Logo após, nas Tabelas 2.3 e 2.4 são apresentados os aminoácidos dispensáveis, com a sua abreviatura e função. E, por fim, na Tabela 2.5 são apresentados os aminoácidos condicionalmente indispensáveis, com a sua abreviatura e função.

2.1.2 A estrutura da proteína

As proteínas são constituídas por apenas vinte tipos de aminoácidos conhecidos, mesmo assim, o número de tipos de proteínas existentes na natureza é extremamente grande. Além do número de aminoácidos variar de setenta a alguns milhares, as diferentes seqüências que esses aminoácidos podem formar são praticamente infinitas (BERG; TYMOCZKO; STRYER, 2004; S. JUNIOR; SASSON, 2003).

Aminoácidos Dispensáveis		
Abrev.	Aminoácido	Função
Ala	Alanina <i>Alanine</i>	é o componente principal do tecido de conexão, elemento intermediário do ciclo glucose-alanina, que permite que os músculos e outros tecidos tirem energia dos aminoácidos e obtenham sistema de imunização. Ajuda a melhorar o sistema imunológico;
Arg	Arginina <i>Arginine</i>	pode aumentar a secreção de insulina, glucagon e GH. Ajuda na reabilitação de ferimentos, na formação de colágeno e estimula o sistema imunológico. É precursor da creatina e do ácido gama amino butírico (GABA, um neurotransmissor do cérebro). Pode aumentar a contagem de esperma e a resposta T-lymphocyte. Vital para o funcionamento da glândula pituitária, deve ser tomada antes de dormir. Ela aumenta a produção do hormônio do crescimento;
Asp	Ácido Aspártico <i>Aspartic Acid</i>	reduz os níveis de amônia depois dos exercícios, auxiliando na sua eliminação, além de proteger o sistema nervoso central. Ajuda a converter carboidratos em energia muscular e a melhorar o sistema imunológico;
Cys Cis	Cisteína <i>Cysteine</i>	em conjunto com outras substâncias, auxilia na desintoxicação do organismo, aumentando a eficiência do processo de recuperação e resistência a doenças. Por isso, ajuda a prevenir danos oriundos do álcool e do tabaco. Estimula a atividade das células brancas no sangue. É a principal fonte de enxofre em uma dieta. Auxilia também no crescimento dos cabelos, unhas e na conservação da pele;
Gln	Glutamina <i>Glutamine</i>	é o aminoácido mais abundante, essencial nas funções do sistema imunológico. Também é importante fonte de energia, especialmente para os rins e intestinos durante restrições calóricas. No cérebro, ajuda a memória e estimula a inteligência e a concentração;
Glu	Ácido Glutâmico <i>Glutamic Acid</i>	precursor da glutamina, prolina, ornitina, arginina, glutathione e gaba, é uma fonte potencial de energia, importante no metabolismo do cérebro e de outros aminoácidos. É conhecido como o "combustível do cérebro". Também é necessário para a saúde do sistema nervoso;

Tabela 2.3: Aminoácidos dispensáveis (1/2)

Aminoácidos Dispensáveis		
Abrev.	Aminoácido	Função
Gly Gli	Glicina <i>Glycine</i>	ajuda na fabricação de outros aminoácidos e é parte da estrutura da hemoglobina e cytocromos (enzimas envolvidas na produção de energia). Tem um efeito calmante e é usado muitas vezes para tratar pessoas maníaco-depressivas e agressivas. Reduz a vontade de comer açúcar. Também é necessário para a conservação da pele e dos tecidos musculares;
Pro	Prolina <i>Proline</i>	é o ingrediente mais importante do colágeno. Essencial na formação de tecido de conexão e músculo do coração, é facilmente mobilizado para energia muscular;
Ser	Serina <i>Serine</i>	importante na produção de energia das células, ajuda a memória e funções do sistema nervoso. Melhora o sistema imunológico, produzindo imunoglobulinas e anticorpos;
Tyr Tir	Tirosina <i>Tyrosine</i>	precursor dos neurotransmissores dopamina, norepinefrina e epinefrina. Aumenta a sensação de bem-estar.

Tabela 2.4: Aminoácidos dispensáveis (2/2)

Aminoácidos Condisionalmente Indispensáveis		
Abrev.	Aminoácido	Função
Arg	Arginina <i>Arginine</i>	pode aumentar a secreção de insulina, glucagon e GH. Ajuda na reabilitação de ferimentos, na formação de colágeno e estimula o sistema imunológico. É precursor da creatina e do ácido gama amino butírico (GABA, um neurotransmissor do cérebro). Pode aumentar a contagem de esperma e a resposta T-lymphocyte. Vital para o funcionamento da glândula pituitária, deve ser tomada antes de dormir. Ela aumenta a produção do hormônio do crescimento;
Cys Cis	Cisteína <i>Cysteine</i>	em conjunto com outras substâncias, auxilia na desintoxicação do organismo, aumentando a eficiência do processo de recuperação e resistência a doenças. Por isso, ajuda a prevenir danos oriundos do álcool e do tabaco. Estimula a atividade das células brancas no sangue. É a principal fonte de enxofre em uma dieta. Auxilia também no crescimento dos cabelos, unhas e na conservação da pele;
Tyr Tir	Tirosina <i>Tyrosine</i>	precursor dos neurotransmissores dopamina, norepinefrina e epinefrina. Aumenta a sensação de bem-estar.

Tabela 2.5: Aminoácidos condicionalmente indispensáveis

As proteínas podem ser estudadas sobre dois enfoques, a constituição do fio protéico e a forma da molécula. A constituição do fio protéico trata dos tipos de aminoácidos que compõem a proteína, ou seja, quando estuda-se uma proteína quanto aos tipos de aminoácidos que fazem parte dela e quanto à seqüência em que estão ordenados, se esta analisando sua estrutura primária. A seqüência exata dos aminoácidos numa proteína é extremamente importante para o desempenho de sua função, quando a célula, por motivos hereditários, comete enganos trocando um aminoácido por outro na seqüência de uma proteína, pode alterar totalmente o funcionamento da proteína, causando doenças sérias ou a proteína fica sem função e desaparece (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

A forma da molécula trata de como a cadeia de aminoácidos se torce, já que as proteínas não são fios esticados, formando uma hélice, como um fio de telefone. Esse enovelamento na forma de uma hélice representa o que os químicos chamam de estrutura secundária. E a própria hélice se torce sobre si mesma, adquirindo uma forma espacial arredondada. A forma definitiva da proteína é determinada pelo modo como a hélice se dobra e é chamada de estrutura terciária. Por razões químicas, a estrutura terciária depende da seqüência de aminoácidos, assim, proteínas com seqüências diferentes têm formas ou estruturas terciárias também diferentes (BERG; TYMOCZKO; STRYER, 2004; S. JUNIOR; SASSON, 2003). A Figura 2.3 mostra as diferentes estruturas da proteína.

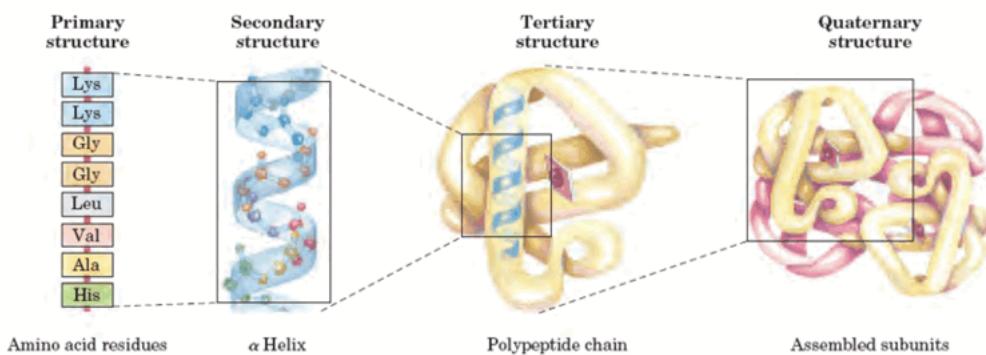


Figura 2.3: Estrutura da proteína
Fonte: (NELSON; COX, 2006)

2.1.3 Proteínas: relação entre forma e função

Em muitas proteínas, a forma determina seu papel biológico, ou seja, proteínas diferentes, tendo formas diferentes, apresentam atividade biológica diferente. Quando uma proteína é submetida a certos tratamentos químicos, ou então a temperaturas elevadas, ela se altera muitas vezes de forma permanente, o que é chamado de desnaturação (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

Isso ocorre quando o tratamento empregado rompe certas ligações químicas que mantinham a forma da molécula e quando as proteínas celulares se deformam, elas perdem a capacidade de desempenhar suas funções (S. JUNIOR; SASSON, 2003).

2.2 O dogma central da biologia molecular

O dogma central da biologia estabelece que o DNA atua como um modelo para se replicar, ele também é transcrito no RNA e o RNA é traduzido em proteína, conforme ilustra a Figura 2.4 (GIBAS; JAMBECK, 2001).

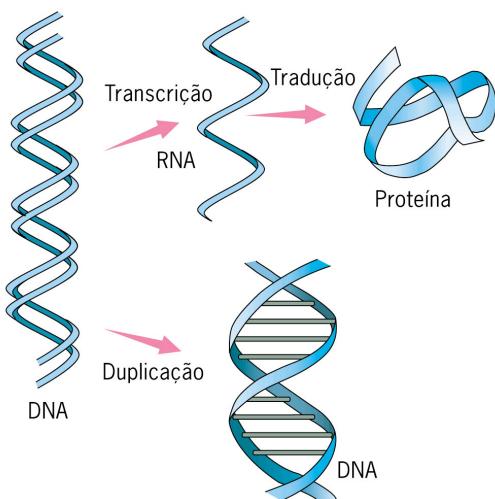


Figura 2.4: Duplicação, transcrição e tradução

Fonte: (S. JUNIOR; SASSON, 2003)

A informação genética é conservada e passada para os descendentes por meio do processo de replicação, e essas informações genéticas também são usadas pelo organismo individual por meio de processos de transcrição e tradução (GIBAS; JAMBECK, 2001; NELSON; COX, 2006).

O DNA genômico contém o plano mestre de um ser vivo e, sem ele, os organismos não seriam capazes de se auto-replicarem. A seqüência de DNA “unidimensional” é só informação, que é lida pelo sistema de síntese da proteína da célula (GIBAS; JAMBECK, 2001; NELSON; COX, 2006).

2.2.1 A estrutura do DNA e do RNA

O DNA e o RNA são macromoléculas constituídas por centenas ou milhares de nucleotídeos ligados entre si. Cada nucleotídeo é sempre composto por três partes: um grupo fosfato (um açúcar do grupo das pentoses), a desoxirribose (no caso do DNA) e uma base nitrogenada (S. JUNIOR; SASSON, 2003; ZAHA, 2001). A Figura 2.5 ilustra essa ligação para os quatro diferentes tipos de nucleotídeos.

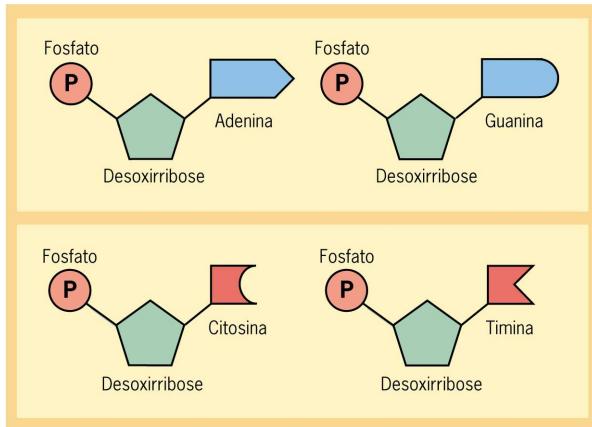


Figura 2.5: Nucleotídeos do DNA
Fonte: (S. JUNIOR; SASSON, 2003)

As bases nitrogenadas podem ser púricas, adenina e guanina ou pirimídicas, citosina e timina, a base uracila, também pirimídica, é encontrada somente no RNA. As bases púricas, maiores, são constituídas de um anel duplo de carbono e nitrogênio, enquanto que as pirimídicas, menores, são compostas por um anel simples (BERG; TYMOCZKO; STRYER, 2004; S. JUNIOR; SASSON, 2003; ZAHA, 2001).

A molécula de DNA é constituída por duas cadeias de nucleotídeos e em cada cadeia, os nucleotídeos estão ligados uns aos outros pelos fosfatos. Na molécula de DNA, as duas cadeias de nucleotídeos estão ligadas uma à outra pelas suas bases nitrogenadas, por meio de pontes de hidrogênio. Por motivos de configuração molecular, a ligação ocorre entre pares de bases específicas, assim, a adenina liga-se a timina, e a citosina liga-se à guanina. A molécula de DNA assemelha-se, então, a uma escada de corda: nela, fosfatos e pentoses representam os corrimãos, enquanto os degraus da escada são representados pelos pares de base (S. JUNIOR; SASSON, 2003; PARIS, 2008). A Figura 2.6 ilustra o esquema de molécula de DNA.

As estruturas do DNA e do RNA se diferem em três aspectos: enquanto o DNA é formado por duas cadeias de nucleotídeos, o RNA é formado por uma fita única; a pentose no RNA é sempre a ribose (no DNA é a desoxirribose); e a uracila é exclusiva do RNA, da mesma forma que a timina caracteriza o DNA (BERG; TYMOCZKO; STRYER, 2004; S. JUNIOR; SASSON, 2003; ZAHA, 2001). A Figura 2.7 ilustra o esquema de molécula de RNA.

2.2.2 Replicação de DNA

A especificidade do pareamento das bases sugere que cada uma das fitas parentais separadas pode atuar como molde para a síntese de uma fita-filha complementar (GIBAS; JAMBECK, 2001; LEWIN, 2001). Isso é demonstrado na Figura 2.8 e o seu funcionamento ocorre da seguinte forma:

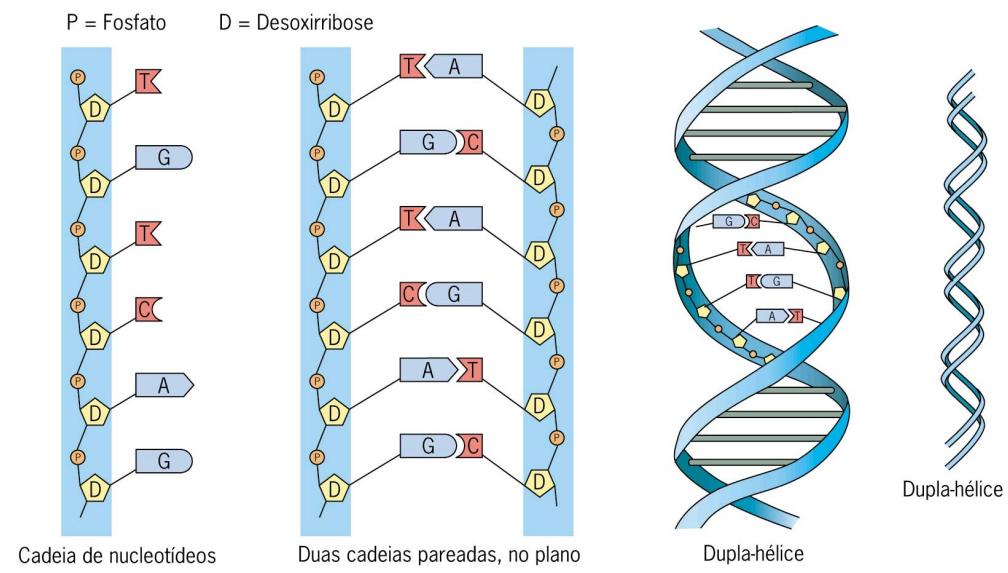


Figura 2.6: Esquema de molécula de DNA
Fonte: (S. JUNIOR; SASSON, 2003)

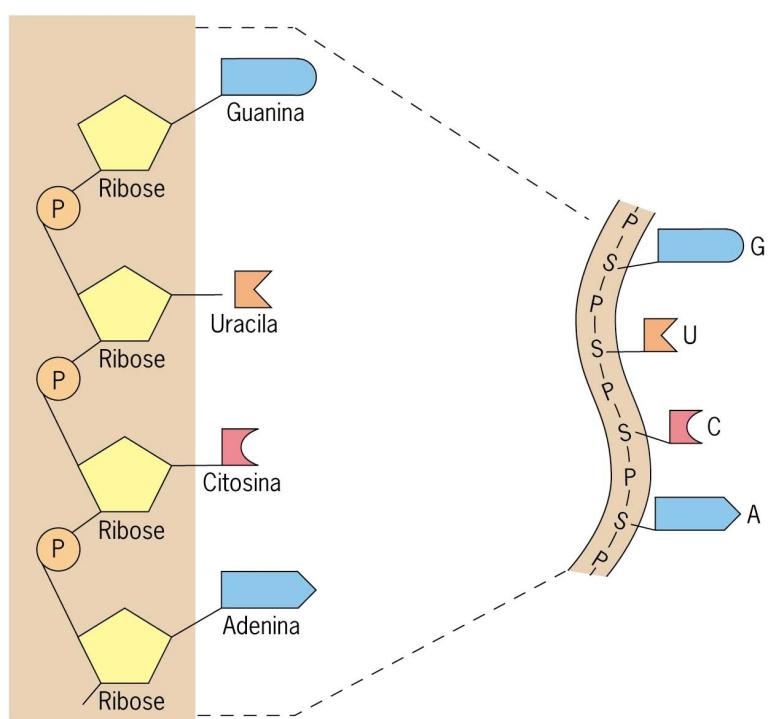


Figura 2.7: Esquema de molécula de RNA
Fonte: (S. JUNIOR; SASSON, 2003)

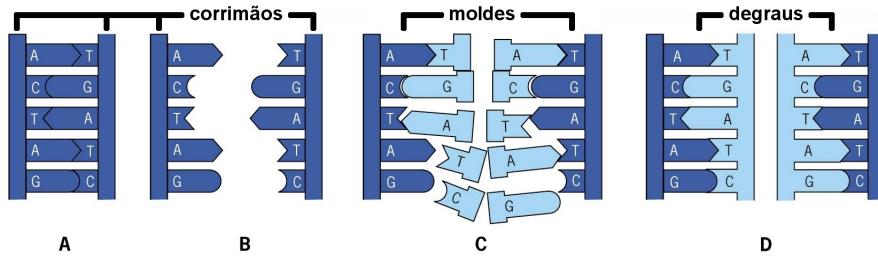


Figura 2.8: Esquema de duplicação de DNA

Fonte: (S. JUNIOR; SASSON, 2003)

- Nessa etapa tem-se, duas fitas complementares de DNA em que as bases nitrogenadas estão ligadas por pontes de hidrogênio. Os “corrimãos” laterais são formados por fosfato e desoxiribose intercalados (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006);
- No primeiro passo da duplicação, ou replicação, as pontes de hidrogênio que ligam as bases nitrogenadas se rompem, e as duas fitas se separam (S. JUNIOR; SASSON, 2003; ZAHA, 2001);
- Cada uma das fitas originais (azul-escuro) serve de “molde” para a produção de fitas novas (azul-claro), que são formadas por nucleotídeos de DNA livres, que já estavam presentes na célula. Os nucleotídeos novos também se ligam entre si, formando um novo “corrimão”, com açúcar e fosfato alternados (GIBAS; JAMBECK, 2001; S. JUNIOR; SASSON, 2003);
- Nessa etapa temos duas moléculas de DNA idênticas quanto a seqüência de pares de bases, de “degraus”. Cada molécula, agora, tem uma fita velha (azul-escuro), que pertenceu à molécula original, e uma fita nova (azul-claro), recém-formada (S. JUNIOR; SASSON, 2003).

Para a duplicação acontecer, são necessárias várias enzimas, uma delas, a helicase que separa as duas hélices, uma outra, a DNA polimerase que permite a ligação de nucleotídeos novos ao molde de DNA (S. JUNIOR; SASSON, 2003).

Esse processo de duplicação em que cada molécula-filha conservou a metade da molécula-mãe, também é chamado de duplicação semiconservativa (S. JUNIOR; SASSON, 2003; LEWIN, 2001; NELSON; COX, 2006). A Figura 2.9 ilustra em três dimensões o processo de duplicação do DNA.

2.2.3 Transcrição de DNA

O DNA não atua somente como um modelo para fazer cópias de si mesmo, mas também como modelo para uma molécula chamada ácido ribonucléico (RNA) (GIBAS; JAMBECK, 2001; LESK, 2008; NELSON; COX, 2006).

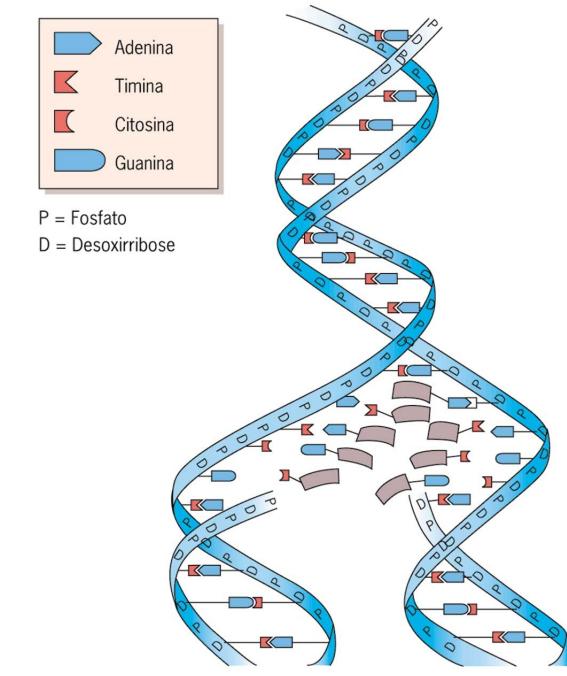


Figura 2.9: Duplicação do DNA
Fonte: (S. JUNIOR; SASSON, 2003)

Enquanto a duplicação, ou replicação, é uma propriedade que permite a transmissão da informação genética às células-filhas, a produção de RNA relaciona-se à síntese de proteínas, no citoplasma (S. JUNIOR; SASSON, 2003). Isso é demonstrado na Figura 2.10 e o seu funcionamento ocorre da seguinte forma:

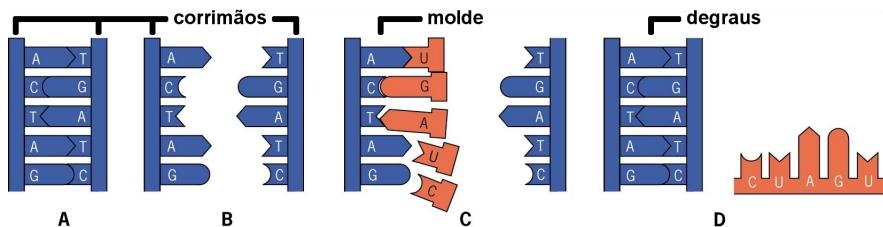


Figura 2.10: Esquema de transcrição de RNA
Fonte: (S. JUNIOR; SASSON, 2003)

- Nessa etapa temos um trecho de molécula de DNA, constituído por duas fitas complementares, cujas bases estão ligadas entre si por pontes de hidrogênio (S. JUNIOR; SASSON, 2003).
- No primeiro passo da síntese de RNA, as pontes de hidrogênio se rompem e as duas fitas se afastam uma da outra (S. JUNIOR; SASSON, 2003).

- c) Apenas uma das fitas de DNA funciona como “molde”. Nela, encaixam-se nucleotídeos de RNA já existente na célula (em vermelho), que têm a ribose como açúcar. Repare que, na adenina do DNA, encaixa-se um nucleotídeo com a base uracila, exclusiva do RNA, em vez de timina, exclusiva do DNA. Os demais tipos de encaixe são semelhantes aos que ocorrem na replicação do DNA. Enquanto isso, a segunda fita de DNA permanece inativa (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).
- d) Uma vez produzida, a fita de RNA se destaca do “molde” de DNA e irá migrar, mais tarde, para o citoplasma. Por fim, as duas fitas de DNA voltam a parear (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

A molécula de RNA é uma fita simples e a informação para que ela seja produzida está contida apenas numa das fitas do DNA, e não nas duas. Para o processo ocorrer, é necessária uma enzima especial chamada de RNA polimerase, que, além de afastar as fitas de DNA, permite o encaixe dos nucleotídeos de RNA (S. JUNIOR; SASSON, 2003). A Figura 2.11 ilustra em três dimensões o processo de transcrição do DNA em RNA.

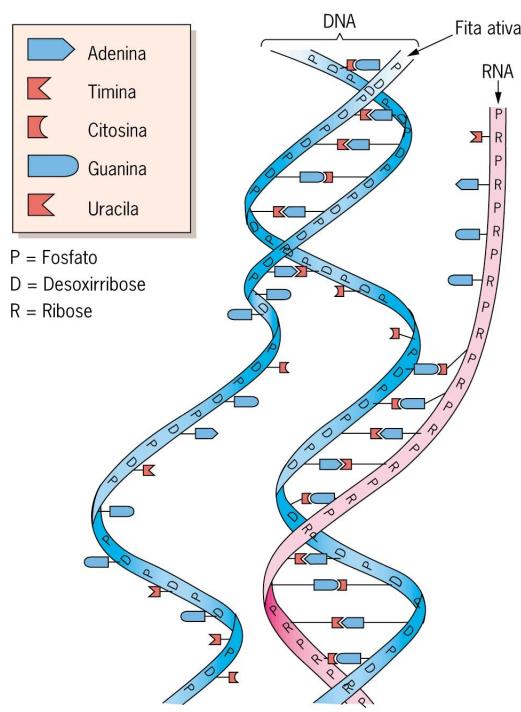


Figura 2.11: Transcrição de DNA em RNA
Fonte: (S. JUNIOR; SASSON, 2003)

O genoma fornece um modelo para a síntese de uma variedade de moléculas de RNA, as três principais são o RNA mensageiro, o RNA transportador e o RNA ribossômico (GIBAS; JAMBECK, 2001; LESK, 2008).

As moléculas de RNA mensageiro (mRNA) são transcritas do RNA dos genes, elas levam informações do genoma para o ribossomo, a maquinária de síntese protéica da célula. As moléculas de RNA transportador (tRNA) são moléculas de RNA não traduzidas que transportam aminoácidos, os blocos de construção das proteínas, para os ribossomos. Finalmente, as moléculas de RNA ribossômico (rRNA) são os componentes de RNA não traduzido dos ribossomos, que são complexos de proteínas e RNA. Os rRNA estão envolvidos na fixação das moléculas de mRNA e na catálise de algumas etapas no processo de tradução (GIBAS; JAMBECK, 2001).

2.2.4 Tradução de mRNA

A tradução do mRNA em proteína é a etapa final na colocação das informações contidas no genoma em funcionamento na célula. Como o DNA, as proteínas são polímeros lineares criados de um alfabeto de unidades quimicamente variáveis, o alfabeto das proteínas é um conjunto de pequenas moléculas denominadas aminoácidos (GIBAS; JAMBECK, 2001; LESK, 2008).

Ao contrário do DNA, a seqüência química de uma proteína tem uma composição físico-químico, assim como um conteúdo informativo. Cada um dos 20 aminoácidos encontrados com mais freqüência nas proteínas tem uma natureza química diferente, determinada por sua cadeia lateral (um grupo químico que varia de aminoácido para aminoácido). A seqüência química da proteína se chama estrutura primária, mas a maneira pela qual a seqüência se dobra para formar uma molécula compacta é tão importante para a função da proteína como é sua estrutura primária. Os elementos das estruturas secundária e terciária que compõem a dobra final da proteína podem juntar partes distantes da seqüência química da proteína para formar sítios funcionais (GIBAS; JAMBECK, 2001; NELSON; COX, 2006).

À correspondência entre trincas de bases do DNA, trincas de bases do RNA e aminoácidos chamamos de código genético. Cada trinca de bases no DNA ou RNA é denominada códon, essas trincas representam “palavras” do código genético e cada “palavra” corresponde a um “objeto”, o aminoácido (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

Como mostra a Tabela 2.6, o código genético é o código que converte o DNA em proteína. Ele utiliza três bases de DNA (códons) para codificar cada aminoácido em uma seqüência de proteína. As combinações simples informam que há 64 maneiras de selecionar 3 nucleotídeos de um conjunto de 4, portanto, há 64 códons possíveis e somente 20 aminoácidos (LESK, 2008).

Alguns códons são redundantes, outros têm a função especial de informar ao mecanismo de tradução da célula para parar de converter uma molécula de mRNA (GIBAS; JAMBECK, 2001). A Figura 2.12 mostra a correspondência entre as unidades de DNA e do RNA e os aminoácidos da proteína a ser sintetizada.

		Segunda Posição									
		U		C		A		G			
Primeira Posição	U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U	Terceira Posição
		UUC		UCC		UAC		UGC		C	
		UUA	Leu	UCA		UAA	Stop	UGA	Stop	A	
		UUG		UCG		UAG	Stop	UGG	Trp	G	
	C	CUU		CCU	Pro	CAU	His	CGU	Arg	U	
		CUC		CCC		CAC		CGC		C	
		CUA		CCA		CAA	Gln	CGA		A	
		CUG		CCG		CAG		CGG		G	
	A	AUU		ACU	Thr	AAU	Asn	AGU	Ser	U	
		AUC		ACC		AAC		AGC		C	
		AUA		ACA		AAA	Lys	AGA	Arg	A	
		AUG	Met	ACG		AAG		AGG		G	
	G	GUU		GCU	Ala	GAU	Asp	GGU	Gly	U	
		GUC		GCC		GAC		GGC		C	
		GUA		GCA		GAA	Glu	GGA		A	
		GUG		GCG		GAG		GGG		G	

Tabela 2.6: Código genético
Fonte: (GIBAS; JAMBECK, 2001)

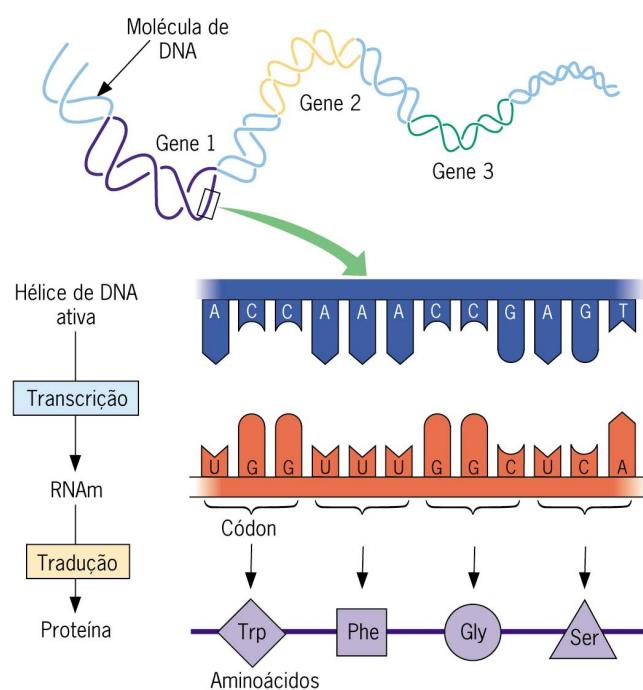


Figura 2.12: Correspondência entre DNA, RNA e Aminoácidos
Fonte: (S. JUNIOR; SASSON, 2003)

2.2.5 Síntese de proteínas

Na Figura 2.13 os aminoácidos foram representados em vermelho, como bolinhas, triângulos, etc., para melhor ilustrar a explicação.

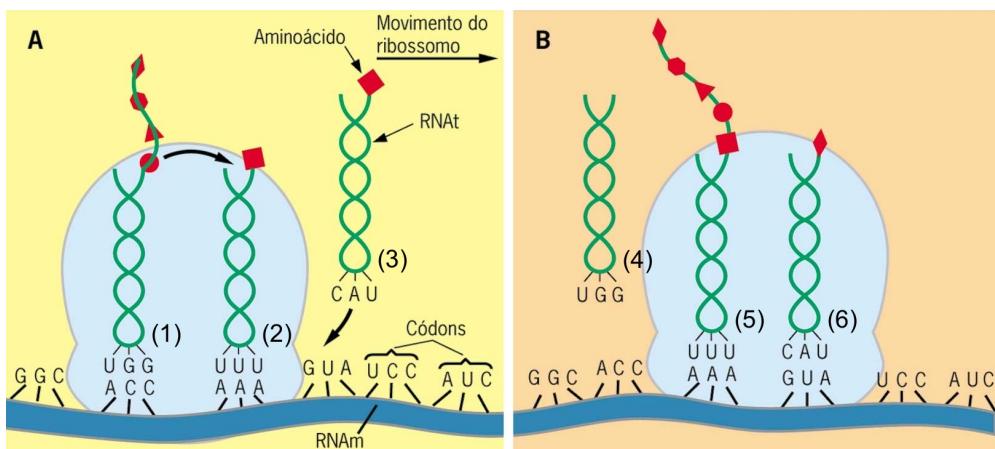


Figura 2.13: Síntese protéica
Fonte: (S. JUNIOR; SASSON, 2003)

O ribossomo do esquema A (azul) desliza ao longo da fita de mRNA, movendo-se da esquerda para direita, no momento ele abrange dois códons do mRNA. O tRNA com o anticôdon UGG (1) está ligado à cadeia de aminoácidos, o segundo tRNA com o anticôdon UUU (2), encaixa-se no códon AAA (2) do mRNA e está trazendo o aminoácido “bolinha”, entre os aminoácidos “bolinha”, o último da cadeia a ser fabricado, e “quadradinho”, que acaba de ser trazido, vai se formar uma ligação peptídica (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

No esquema B é exemplificado o que ocorre na seqüência, a ligação peptídica entre os aminoácidos completou-se, e a cadeia polipeptídica foi acrescida de um aminoácido. O tRNA com o anticôdon UGG (4) então desliga-se da cadeia de aminoácidos e volta para o citoplasma, podendo buscar um novo aminoácido “bolinha”. A proteína em formação está ligada agora ao tRNA com o anticôdon UUU (5), o ribossomo deslizou para a direita, abrangendo um novo códon do mRNA, GUA (6). O tRNA com o anticôdon CAU (6), o único que pode se encaixar, está trazendo o aminoácido “losango”. Logo haverá ligação peptídica entre “quadradinho” e “losango” e o penúltimo tRNA, UUU (5), se desligará, o ribossomo deslizará para a direita, abrangendo mais um códon, e assim por diante (S. JUNIOR; SASSON, 2003).

A cada códon que o ribossomo abrange, é acrescentado um aminoácido específico à proteína em crescimento, quando o ribossomo tiver percorrido todo o mRNA, toda a mensagem terá sido lida e a proteína estará pronta. Então o ribossomo se desliga do mRNA. A mesma fita de RNA pode ser lida por vários ribossomos e cada um deles produzirá uma molécula de proteína exatamente igual (S. JUNIOR; SASSON, 2003; NELSON; COX, 2006).

2.2.6 Genomas e genes

A seqüência completa de DNA que codifica um ser vivo é chamada de genoma, mas o genoma não funciona como uma seqüência longa, ele é dividido em genes individuais (GIBAS; JAMBECK, 2001). Um gene é uma seqüência de DNA necessária para a síntese de uma proteína, porém existem ao longo dos cromossomos algumas seqüências de DNA especializadas capazes de transcrever, mas que não contêm informação para a síntese de proteína (S. JUNIOR; SASSON, 2003).

Existem três tipos de genes. Os genes codificadores de proteínas são modelos para gerar moléculas de proteínas. Cada proteína codificada pelo genoma é uma máquina química com um propósito distinto no organismo. Os genes especificadores de RNA também são modelos para as máquinas químicas, mas os blocos criadores das máquinas de RNA são diferentes dos que compõem a proteína. E, por fim, os introns (genes não transcritos) são regiões do DNA genômico que possuem algum propósito funcional, mas não alcançam esse propósito, sendo transcritos ou convertidos para criar outra molécula (GIBAS; JAMBECK, 2001; LESK, 2008).

2.2.7 Evolução molecular

Os erros na replicação e transcrição de DNA são relativamente comuns. Se esses erros ocorrem nas células reprodutoras de um organismo, eles podem ser transmitidos aos seus descendentes. As alterações na seqüência de DNA são conhecidas como mutações, e essas mutações podem ter resultados prejudiciais, resultados que diminuem a probabilidade de sobrevivência dos descendentes até a idade adulta, resultados benéficos ou ser neutras. Se uma mutação não mata o organismo antes que ele se reproduza, a mutação pode se fixar na população depois de muitas gerações, a lenta acumulação dessas mudanças é responsável pelo processo conhecido como evolução (GIBAS; JAMBECK, 2001; S. JUNIOR; SASSON, 2003).

O acesso as seqüências de DNA permite o acesso a um melhor entendimento da evolução. Nossa entendimento do mecanismo de evolução molecular como um processo gradual de acumulação de mutações de seqüências de DNA é a justificativa para o desenvolvimento de hipóteses baseadas na comparação das seqüências de DNA e de proteínas (GIBAS; JAMBECK, 2001).

2.3 Bancos de dados biológicos

Nas subseções que seguem será comentada um pouco da história e qual a função de alguns dos principais bancos de dados biológicos existentes.

2.3.1 DDBJ

O DDBJ¹ (*DNA Data Bank of Japan*) iniciou suas atividades em banco de dados de DNA em 1986 no *National Institute of Genetics* (NIG), com o aval do Ministério da Educação, Ciência, Esporte e Cultura. Desde o início, o DDBJ tem funcionado com uma das bases de dados internacional de DNA, que inclui EBI (*European Bioinformatics Institute*, responsável pela base de dados EMBL) na Europa e NCBI (*National Center for Biotechnology Information*, responsável pela base de dados GenBank) nos Estados Unidos da América como os dois outros membros. Conseqüentemente o DDBJ tem colaborado com os outros dois bancos de dados através do intercâmbio de dados e informações pela Internet e pela sua regular participação nas duas reuniões, a *International DNA Data Banks Advisory Meeting* e a *International DNA Data Banks Collaborative Meeting*.

O *Center for Information Biology* no NIG foi reorganizado como o *Center for Information Biology and DNA Data Bank of Japan* (CIB-DDBJ) em 2001. O novo centro desempenha um papel importante na realização de projetos de investigação em informações biológicas. O DDBJ é o único banco de dados de DNA no Japão que é oficialmente certificado para recolher seqüências de DNA de pesquisadores, e de emitir o número reconhecido internacionalmente para adesão dos dados fornecidos. O DDBJ coleta dados principalmente de pesquisadores japoneses, mas evidentemente também aceita dados e emite o número de adesão aos investigadores em todos os outros países. A troca de dados coletados entre a EMBL, EBI e GenBank, NCBI em uma base diária, proporciona que os três bancos partilhem de praticamente os mesmos dados em um determinado momento. O CIB-DDBJ também fornece muitas ferramentas para a recuperação de dados e análises desenvolvidas através do DDBJ.

2.3.2 EMBL-EBI

O EBI² (*European Bioinformatics Institute*) é parte integrante do EMBL (*European Molecular Biology Laboratory*). O EMBL-EBI foi o primeiro banco de dados do mundo de seqüência nucleotídica tendo surgido em 1980 em Heidelberg, na Alemanha. O que começou com uma modesta tarefa de abstrair informações da literatura, logo se tornou uma importante base de dados com necessidade de pessoal altamente qualificado em informática com o início do projeto genoma. Seus grupos de investigação visam compreender a biologia através do desenvolvimento de novas abordagens para a interpretação dos dados biológicos.

¹DDBJ - DNA Data Bank of Japan. Disponível em: <<http://www.ddbj.nig.ac.jp>>. Acesso em: 14 de maio de 2009

²EMBL-EBI. Disponível em: <<http://www.ebi.ac.uk>>. Acesso em: 30 de abril de 2009

2.3.3 GenBank

O GenBank³ é um banco de dados de seqüências genéticas e uma coleção de todas as anotações de seqüências de DNA disponíveis. Essa base contém aproximadamente 85.759.586.764 bases em 82.853.685 registros de seqüências. Os lançamentos de novas versões são feitos a cada dois meses. O GenBank é parte do *International Nucleotide Sequence Database Collaboration*, que inclui o DNA DataBank do Japão (DDBJ), o European Molecular Biology Laboratory (EMBL) e o GenBank do NCBI.

A base de dados do GenBank é projetada para proporcionar e incentivar o acesso pela comunidade científica a mais atualizada e completa seqüência de informações de DNA. O NCBI está continuamente desenvolvendo novas ferramentas e atualizando as já existentes para melhorar a apresentação e o acesso ao GenBank. O NCBI não coloca restrições à utilização ou distribuição dos dados do GenBank.

2.3.4 OMIM

OMIM⁴ (*Online Mendelian Inheritance in Man*) é um catálogo de atualização contínua de genes humanos e doenças genéticas (que é o seu principal foco), com *links* para literaturas de referência, seqüências de registros, mapas, dados e afins. OMIM está baseado no texto *Mendelian Inheritance in Man*, de autoria do Dr. Victor A. McKusick e uma equipe de redatores e editores científicos no John Hopkins University e da população.

Esta base de dados foi iniciada no início dos anos de 1960 pelo Dr. Victor A. McKusick com um catálogo de traços mendelianos e transtornos, intitulado *Mendelian Inheritance in Man* (MIM). Doze edições do livro foram publicadas entre 1966 e 1998. A versão online, OMIM, foi criada em 1985 por uma colaboração entre a *National Library of Medicine* e o *William H. Welch Medical Library* em Johns Hopkins e em 1987 foi disponibilizada na Internet. Em 1995, OMIM foi desenvolvido para *World Wide Web* pelo NCBI (*National Center for Biotechnology Information*).

2.3.5 PDB

O PDB⁵ (*Protein Data Bank*) é o único repositório de informações sobre as estruturas 3D de grandes moléculas biológicas, incluindo proteínas e ácidos nucléicos. Compreender a forma de uma molécula ajuda a compreender como ela funciona, esse conhecimento pode ser usado para ajudar a deduzir o seu papel na estrutura da saúde humana e doença, e desenvolvimento de drogas.

³GenBank at NCBI. Disponível em: <<http://www.ncbi.nlm.nih.gov/Genbank>>. Acesso em: 14 de maio de 2009

⁴OMIM - Online Mendelian Inheritance in Man. Disponível em: <<http://www.ncbi.nlm.nih.gov/omim>>. Acesso em: 24 de abril de 2009

⁵PDB - Protein Data Bank. Disponível em: <<http://www.rcsb.org/pdb/home/home.do>>. Acesso em: 14 de maio de 2009

O PDB foi criado em 1971 no *Brookhaven National Laboratory* e inicialmente continha sete estruturas. Em 1998, o *Research Collaboratory for Structural Bioinformatics* (RCSB) ficou responsável pela gestão do PDB. Em 2003, o wwpPDB foi formado para manter um único arquivo do PDB de estruturas de dados macromoleculares que é livre e publicamente disponíveis para a comunidade global. O wwpPDB é constituído por organizações que atuam como deposição, processamento de dados e centros de distribuição para os dados PDB. Além disso, o RCSB PDB suporta um site onde os visitantes podem realizar consultas simples e complexas sobre os dados, analisar e visualizar os resultados. O RCSB PDB está localizado na Rutgers, *The State University of New Jersey* e na *University of California*, San Diego.

O RCSB PDB é um membro da wwpPDB, um esforço de colaboração com PDBe (Reino Unido), PDBj (Japão), e BMRB (E.U.A.) para assegurar que o arquivo PDB seja global e uniforme. O PDB arquivo está disponível sem nenhum custo para os utilizadores e novas estruturas são liberadas semanalmente.

2.3.6 STRING

STRING⁶ é uma base de dados em que constam interações de proteínas conhecidas e previsíveis, essas interações podem ser associações diretas (físicas) e indiretas (funcionais) e são provenientes de quatro fontes: contexto genômico, experimentos, coexpressão e conhecimentos prévios.

2.3.7 Swiss-Prot

O Swiss-Prot⁷ é um banco de dados de seqüência de proteínas que se empenha em oferecer um elevado nível de anotação (como a descrição da função de uma proteína, suas estruturas de domínios, variantes, etc.), um nível mínimo de redundância e um alto nível de integração com outras bases de dados. Esse banco de dados é desenvolvido pelo *Swiss-Prot group* no *Swiss Institute of Bioinformatics* (SIB) e no *European Bioinformatics Institute* (EBI).

2.4 Considerações finais

Nesse capítulo foi explicado a função das proteínas e o dogma central da biologia molecular, também foram apresentados alguns exemplos de repositórios de informações biológicas, quando possível, dando ênfase ao conteúdo que eles disponibilizam. No próximo capítulo será apresentado o fluxo de pesquisa de uma doença gênica, que é o foco desse trabalho, e será feita uma abordagem mais voltada para a área de biologia de sistemas.

⁶STRING. Disponível em: <<http://string.embl.de>>. Acesso em: 30 de abril de 2009

⁷Swiss-Prot. Disponível em: <<http://www.expasy.ch/sprot>>. Acesso em: 14 de maio de 2009

3 BIOLOGIA DE SISTEMAS

Após o Projeto Genoma Humano ter completado sua fase inicial, cientistas e políticos começaram a articular cada vez mais, visões de como a tecnologia orientada para a aquisição de conhecimento genômico poderia ser transformada em estratégias de intervenção. A área em que muitas destas ambições e esperanças convergiram é agora o que é chamado de biologia de sistemas. O objetivo global da biologia de sistemas é o objetivo final da biologia moderna, a obtenção de uma fundamental, abrangente e sistemática compreensão da vida. Para atingir esta meta, existe a intenção de integrar sistemas de biólogos para obter uma explicação global para DNA, RNA, proteínas e dados metabólicos, combinando modelagem matemática e uma extensa análise computacional (O'MALLEY; DUPRE, 2005).

A biologia de sistema não prevê transformar as compreensões e práticas dos biólogos, mas os seus métodos e conceitos prevêem ter efeitos importantes sobre outras ciências, como física, engenharia, matemática e ciências sociais. Existem fortes argumentos que a biologia de sistemas é mais do que apenas uma extensão do genoma e da bioinformática, ela é qualitativamente diferente do que já foi alcançado e achado pelas ferramentas atuais (O'MALLEY; DUPRE, 2005).

Nas seções que seguem será apresentado um cenário exemplo de aplicação em biologia de sistemas, a possibilidade de análise de uma doença através de redes de interação de proteínas e o particionamento dos processos biológicos através da ontologia gênica.

3.1 Redes de livre-escala

Uma rede (grafo) é uma coleção de pontos aonde estes pontos são chamados de nodos ou vértices, e os arcos que conectam estes pontos são chamados de arestas. Redes biológicas, representações de relacionamentos biológicos, são construídas para descrever vários fenômenos biológicos. Estas redes variam desde redes que descrevem condutores bioquímicos da célula até redes de mais alto nível tais como redes de neurônios (BEBEK; YANG, 2007).

A conduta de muitos sistemas complexos, das células a Internet, emerge de uma atividade orquestrada de muitos componentes que interagem com outros através de interações aos pares. Em um nível abstrato muito alto, os componentes podem ser reduzidos para uma série de nodos que são conectados por ligações que representam as interações entre dois componentes. Os nodos e ligações juntos formam uma rede, ou, em uma linguagem matemática formal, um grafo (BARABASI; OLTVAI, 2004).

Estabelecer a identidade de várias redes celulares não é trivial, dependendo da natureza das interações, as redes podem ser direcionais ou não direcionais. Em redes direcionais, as interações entre dois nodos têm uma direção bem definida, por exemplo, a direção do fluxo de material de um substrato para um produto em uma reação metabólica. Em redes não-direcionais, as ligações não têm uma direção assinalada, por exemplo, em uma rede de interação de proteínas, uma ligação representa uma relação mútua de amarração bilateral, se a proteína A amarra-se a proteína B, então a proteína B também se amarra a proteína A (BARABASI; OLTVAI, 2004).

A origem da topologia de livre-escala em redes complexas pode ser reduzida a dois mecanismos básicos: crescimento e ligação principal. Crescimento significa que a rede emerge através da ligação de nodos subsequentes, tais como, um novo nodo que é adicionado na rede. Ligação principal significa que novos nodos preferem se ligar aos nodos mais conectados (BARABASI; OLTVAI, 2004).

Uma das principais características, denominada conexão preferencial ou ligação principal, é a tendência de um novo vértice se conectar a um vértice da rede que tem um grau elevado de conexões. Essa característica implica em redes com poucos vértices altamente conectados, denominados *hubs*, e muito vértices com poucas conexões, como mostra a Figura 3.1 (BARABASI; OLTVAI, 2004).

A relação entre a topologia de uma rede biológica e suas propriedades funcionais e evolucionárias sugerem a maioria das redes biológicas, são redes de livre-escala: redes de sinalização, redes celulares, redes metabólicas e redes de interação de proteínas (SIEGAL; PROMISLOW; BERGMAN, 2007).

3.1.1 Redes de sinalização

Redes de sinalização (*signaling networks*) são complexas em termos de eventos químicos e biofísicos e um grande número de interações. A descrição quantitativa nos modelos facilita o mapeamento entre diferentes tipos de métodos de análise para sistemas complexos. Métodos de análise de sistemas podem ressaltar estados estáveis da rede de sinalização e descrever as transições entre eles. Modelos também revelam funcionalidades similares entre propriedades das redes de sinalização e outros sistemas bem-compreendidos, tais como, dispositivos eletrônicos e redes neurais (BHALLA, 2003).

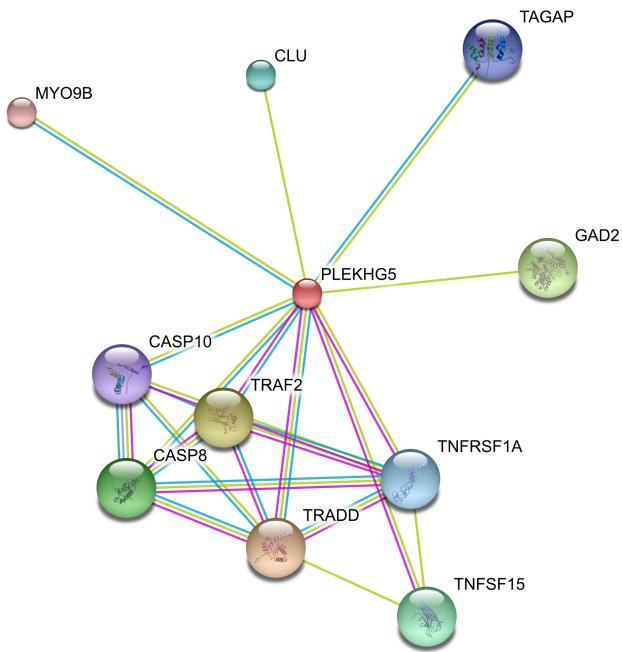


Figura 3.1: Rede de livre-escala
Fonte: (BARABASI; OLTVAI, 2004)

É possível considerar redes de sinalização como sistemas que decodificam entradas complexas em tempo, espaço e química em padrões combinatórios de saída de atividades de sinalização. A combinação de métodos de modelos de computação para capturar a complexidade e detalhes, e abstrações úteis reveladas por estes modelos, é necessário para alcançar uma descrição rigorosa tão boa quanto à compreensão humana (BHALLA, 2003).

3.1.2 Redes celulares

As redes celulares (*cellular networks*) são redes de livre-escala. A primeira evidência desta afirmação surgiu da análise do metabolismo, no qual os nodos são o resultado da atividade metabólica e as ligações representam reações bioquímicas de catalise enzimática. Assim como as reações são irreversíveis, redes metabólicas são direcionadas (BARABASI; OLTVAI, 2004).

3.1.3 Redes metabólicas

Uma rede metabólica (*metabolic networks*) é uma rede de caminhos aonde substratos e produtos metabólicos são conectados com arestas dirigidas. Estes arcos indicam atos de reações metabólicas sobre um determinado substrato e produz um determinado produto. Estudar redes metabólicas permite compreender os mecanismos moleculares de um organismo específico, como por exemplo, glicólise, ciclo de Krebs, etc (BEBEK; YANG, 2007).

3.1.4 Redes de interação de proteínas

A topologia de livre-escala é também, aparentemente, uma característica das redes de interação de proteínas (*protein-protein interaction network*), embora as limitações nos dados sejam substanciais. Redes de interação de proteínas são determinadas, primeiramente, pela análise de duas leveduras híbridas, as quais não provam interações nativas (SIEGAL; PROMISLOW; BERGMAN, 2007).

As ontologias servem para classificar as redes conforme suas funções biológicas, que podem ser componente celular, processo biológico e função molecular.

3.2 Ontologia gênica

O projeto Ontologia Gênica¹ (*Gene Ontology* ou GO) é uma das principais iniciativas na bioinformática e também um esforço de colaboração para dar resposta à necessidade de coerência na descrição dos produtos de gene em diferentes bases de dados. O projeto também é parte de um esforço maior para classificação, o *Open Biomedical Ontologies* (OBO) (FERRO, 2008).

Existem três aspectos distintos para este esforço: em primeiro lugar, escrever e manter as ontologias (vocabulário controlado de gene e propriedades produto de gene) em si, em segundo lugar, fazer ligações cruzadas entre as ontologias e os genes e produtos de gene buscando difundir e assimilar as anotações de dados; e em terceiro lugar, desenvolver ferramentas que facilitam a criação, manutenção e uso de ontologias. Atualmente o projeto GO está organizado em três princípios: componente celular, que é um componente de uma célula, mas com a ressalva de que é parte de um objeto maior que pode ser uma estrutura anatômica; processo biológico, que é uma série de eventos que é realizada por um ou mais conjuntos de funções moleculares ordenadas; e função molecular, que descreve atividades, tais como catalisadores, por exemplo, que ocorrem ao nível molecular. Esses três princípios ou áreas são considerados independentes umas das outras (FERRO, 2008).

O projeto GO foi originalmente constituído em 1998 por um consórcio de investigadores dedicados a estudar o genoma de três organismos modelo: *Drosophila melanogaster* (mosca das frutas), *Mus musculus* (rato), e *Saccharomyces cerevisiae* (levedura). Muitas outras bases de organismo modelo aderiram ao projeto formando o consórcio GO (*GO Consortium*), que é o conjunto de grupos envolvidos ativamente no projeto GO, contribuindo não só com anotação de dados, mas também para o desenvolvimento de ontologias e ferramentas para visualizar e aplicar os dados.

¹The Gene Ontology Project. Disponível em: <<http://www.geneontology.org>>. Acesso em: 24 de abril de 2009

Em janeiro de 2008, o projeto GO já continha mais de 24.500 termos aplicáveis a uma ampla variedade de organismos biológicos. Atualmente existe um conjunto significativo de literatura sobre o desenvolvimento e a utilização do projeto GO e o mesmo já se tornou uma ferramenta padrão no arsenal da bioinformática.

3.3 Fluxo de pesquisa de uma doença

O fluxo de pesquisa para geração das redes de interação da proteína pode seguir dois fluxos, para tanto os mesmos serão descritos separadamente nas seções seguintes usando como base a Figura 3.2.

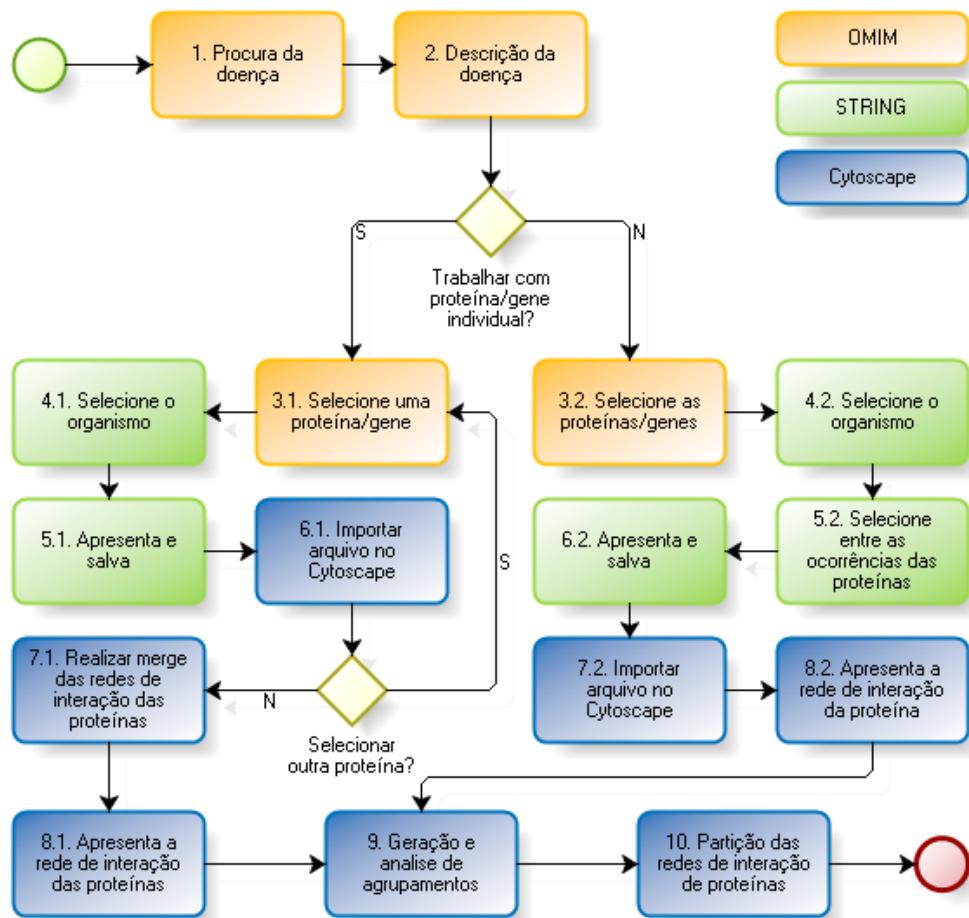


Figura 3.2: Fluxo de pesquisa

3.3.1 Descrição do Fluxo A

O Fluxo A de pesquisa ocorre da seguinte forma. Primeiramente acesse o *site* do OMIM, em <http://www.ncbi.nlm.nih.gov/omim>, digite na caixa de texto ao lado do *label* “for” a doença que procura, como mostra a Figura 3.3, e depois clique no botão “Go” (Atividade 1).

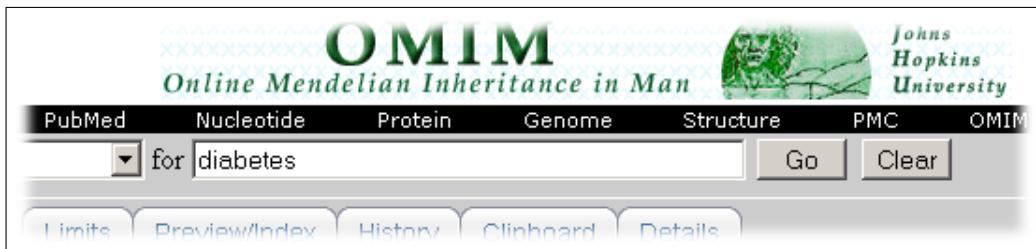


Figura 3.3: Pesquisa da doença

Então o *site* lhe apresentará uma lista com as ocorrências da doença para que você selecione a que você está procurando, como mostra a Figura 3.4.

<input type="checkbox"/> 1: #6222100	Links
DIABETES MELLITUS, INSULIN-DEPENDENT; IDDM	
DIABETES MELLITUS, INSULIN-DEPENDENT, 1, INCLUDED; IDDM1, INCLUDED	
Gene map locus 12q24.2, 12q24.2, 1p13, 7p21, 6p21.3, 6pter-p21, Xp11.23-q13.3	
<input type="checkbox"/> 2: #125853	Links
DIABETES MELLITUS, NONINSULIN-DEPENDENT; NIDDM	
INSULIN RESISTANCE, SUSCEPTIBILITY TO, INCLUDED	
Gene map locus 19p13.2, 17q25, 17q12, 17p13, 15q21-q23, 13q34, 13q12.1, 12q24.2, 11p12-p11.2, 11p15.1, 11p15.1, 10q25.3, 8q24.11, 7q32, 7p15-p13, 7p21, 6q22-q23, 6p22.3, 5q34-q35.2, 4p16.1, 3q28, 3p25, 2q36, 2q32, 2q24.1, 2019p13.2, 17q25, 17q12, 17p13, 15q21-q23, 13q34, 13q12.1, 12q24.2, 11p12-p11.2, 11p15.1, 11p15.1, 10q25.3, 8q24.11, 7q32, 7p15-p13, 7p21, 6q22-q23, 6p22.3, 5q34-q35.2, 4p16.1, 3q28, 3p25, 2q36, 2q32, 2q24.1,	
<input type="checkbox"/> 3: #606176	GeneTests, Links
DIABETES MELLITUS, PERMANENT NEONATAL; PNDM	
DIABETES MELLITUS, PERMANENT NEONATAL, WITH NEUROLOGIC FEATURES, INCLUDED	
Gene map locus 11p15.1, 11p15.1, 7p15-p13	

Figura 3.4: Lista de ocorrências da doença

Após escolhida a ocorrência da doença, o *site* lhe apresentará um relatório com a descrição completa da mesma, como mostra a Figura 3.5 (Atividade 2).

Então localize e selecione uma ou mais proteínas/genes (no exemplo, sublinhadas em vermelho) no relatório, como mostra a Figura 3.6 (Atividade 3.1).

Após selecionada a proteína, acesse o *site* do STRING, em <http://string.embl.de>, e digite-a na caixa de texto localizada abaixo do label “protein name.” na aba “search by name”, como mostra a Figura 3.7 e clique no botão “GO !”.

O *site* lhe apresentará uma lista de organismos, como mostra a Figura 3.8, selecione o que deseja utilizar e clique no botão “Continue →” para prosseguir para a próxima etapa (Atividade 4.1).

Então lhe será apresentada a rede de interação da proteína, como mostra a Figura 3.9, clique no botão “save” para prosseguir (Atividade 5.1).

Após isso o *site* lhe apresentará uma última tela solicitando que você escolha o tipo de arquivo que deseja salvar, como mostra a Figura 3.10, selecione o arquivo do tipo XML (no exemplo, circulado em vermelho).

%222100
DIABETES MELLITUS, INSULIN-DEPENDENT; IDDM

[Links](#)

Alternative titles; symbols

DIABETES MELLITUS, TYPE I
 JUVENILE-ONSET DIABETES; JOD
 DIABETES MELLITUS, INSULIN-DEPENDENT, 1, INCLUDED; IDDM1, INCLUDED
 INSULIN-DEPENDENT DIABETES MELLITUS 1, INCLUDED

Gene map locus [12q24.2, 12q24.2, 1p13, 7p21, 6p21.3, 6pter-p21, Xp11.23-q13.3](#)

TEXT

DESCRIPTION

The type of diabetes mellitus called IDDM is a disorder of glucose homeostasis that is characterized by susceptibility to ketoacidosis in the absence of insulin therapy. It is a genetically heterogeneous autoimmune disease affecting about 0.3% of Caucasian populations ([Todd, 1990](#)). Genetic studies of IDDM have focused on the identification of loci associated with increased susceptibility to this multifactorial phenotype. ☺

Figura 3.5: Relatório da doença

CLINICAL FEATURES

The term diabetes mellitus is not precisely defined and the lack of a consensus on diagnostic criteria has made its genetic analysis difficult. Diabetes mellitus is classified clinically into 2 major forms of the primary illness, insulin-dependent diabetes mellitus (IDDM) and noninsulin-dependent diabetes mellitus (NIDDM; [125853](#)), and secondary forms related to gestation or medical disorders. ☺

Appearance of the IDDM phenotype is thought to require a predisposing genetic background and interaction with other environmental factors. [Rotter and Rimoin \(1978\)](#) hypothesized that there are at least 2 forms of IDDM: a B8 (DR3)-associated form characterized by pancreatic autoimmunity, and a B15-associated form characterized by antibody response to exogenous insulin. Interestingly, the DR3 and DR4 alleles seem to have a synergistic effect on the predisposition to IDDM based on the greatly increased risk observed in persons having both the B8 and B15 antigens ([Sveigaard and Ryder, 1977](#)). [Rotter and Rimoin \(1979\)](#) hypothesized a combined form. [Tolins and Raji \(1988\)](#) cited clinical and experimental evidence to support the idea that those IDDM patients in whom diabetic nephropathy (see [603933](#)) eventually develops may have a genetic predisposition to essential hypertension. ☺

Figura 3.6: Localizando proteínas/genes no relatório

search by name search by protein sequence multiple names multiple sequences

protein name: (examples: #1 #2 #3)
B15

(STRING understands a variety of protein names and accessions; you can also try a [random entry](#))

organism:
auto_detect ▾

interactors wanted:

COGs Proteins Reset GO !

Figura 3.7: Pesquisa da proteína

There are several proteins named 'B15'.
Please select one from the list below and press Continue to proceed.

[← Back](#) [Continue →](#)

organism	protein
<input checked="" type="radio"/> Homo sapiens	NDUFB4 - subunit 4 (EC 1.6.5.3) (EC 1.6.99.3) (NADH-ubiquinone oxidoreductase B15 subunit) (Complex I- B15) (CI- B15)
<input type="radio"/> <i>Bos taurus</i>	NDUFB4 - subunit 4 (EC 1.6.5.3) (EC 1.6.99.3) (NADH-ubiquinone oxidoreductase B15 subunit) (Complex I- B15) (CI- B15)
<input type="radio"/> <i>Arabidopsis thaliana</i>	PP2B15 - F-box protein PP2- B15 (Protein PHLOEM PROTEIN 2-LIKE B15) (AtPP2-B15)
<input type="radio"/> <i>Gallus gallus</i>	NDUFB4 - NADH-ubiquinone oxidoreductase B15 subunit) (Complex I- B15) (CI- B15) (Hypothetical protein Walter) (GGHPW)
<input type="radio"/> <i>Pan troglodytes</i>	NDUFB4 - subunit 4 (EC 1.6.5.3) (EC 1.6.99.3) (NADH-ubiquinone oxidoreductase B15 subunit) (Complex I- B15) (CI- B15)
<input checked="" type="radio"/> Homo sapiens	ENSP00000341045 - UDP glycosyltransferase 2 family, polypeptide B15
<input type="radio"/> <i>Oryzias latipes</i>	ENSORLP00000000568 - UDP glycosyltransferase 2 family, polypeptide B15

[← Back](#) [Continue →](#)

Figura 3.8: Lista de organismos que possuem a proteína

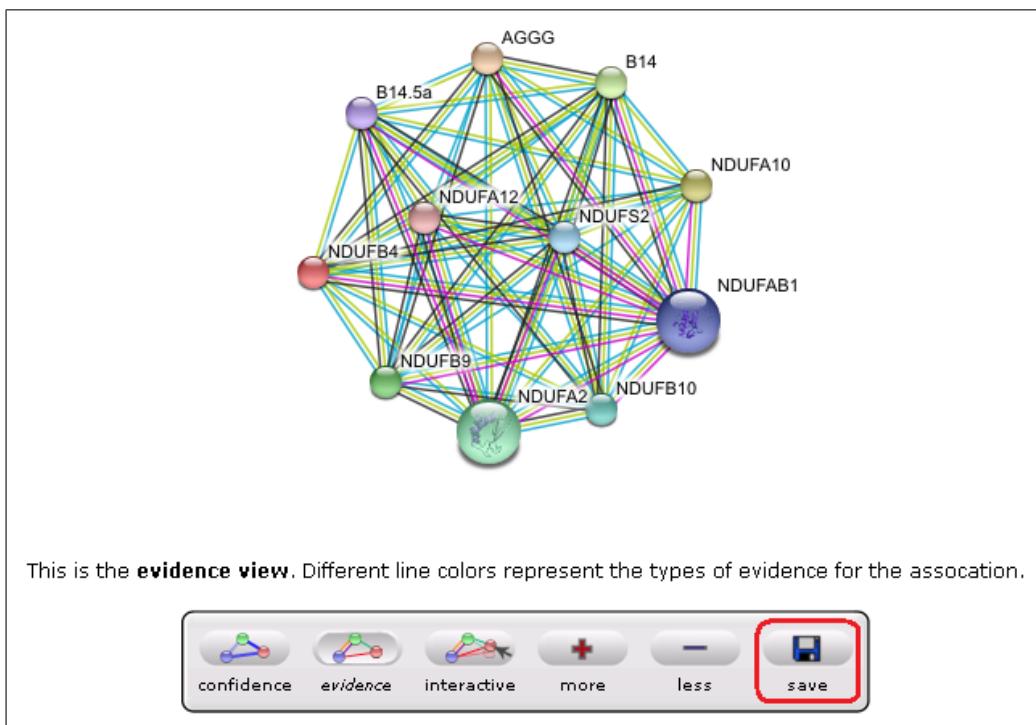


Figura 3.9: Apresentação da rede de interação da proteína

dl_net_image_e_hjJ5Z7GvxxwAk.png.svg	SVG Image: Evidence View (SVG - Scalable Vector Graphics)
xml_summary.hjJ5Z7GvxxwAk.xml	XML Summary (PSI - Proteomics Standards Initiative)
network_medusa.hjJ5Z7GvxxwAk.dat	Graph Layout (Data for the 'Medusa' Network Viewer)

Figura 3.10: Seleção do tipo de arquivo da rede de interação

Uma vez que o arquivo esteja salvo em sua máquina local, abra o software “Cytoscape”, disponível em <http://www.cytoscape.org>, e abra o menu “File → Import → Network (multiple file types)...” (Atividade 6.1).

Selecione o arquivo que deseja importar clicando no botão “Select” e o localizando, então clique no botão “Import” para importar o arquivo, como mostra a Figura 3.11. Repita esse processo com quantas proteínas você desejar.

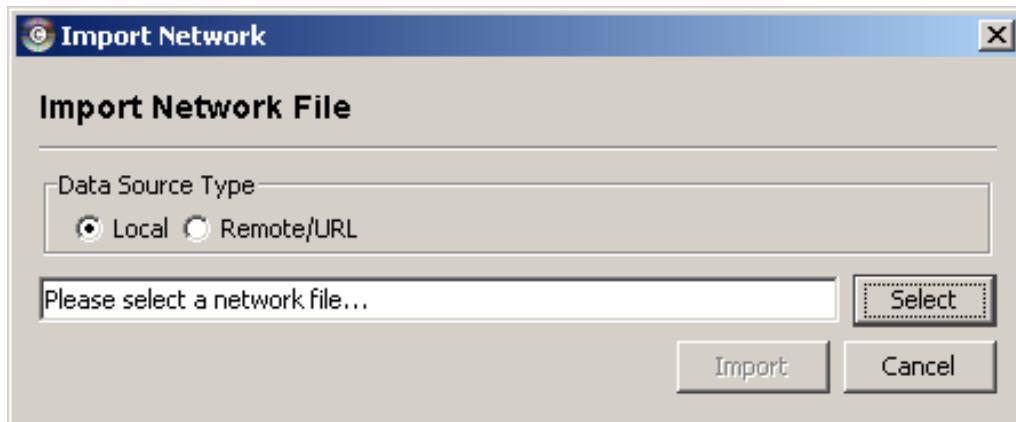


Figura 3.11: Importando rede de interação da(s) proteína(s)

Uma vez que todos os arquivos que deseja utilizar tenham sido importados, você poderá realizar um “merge” das redes de interações. Abra o menu “Plugins → Merge networks”, selecione as redes que deseja realizar o “merge”, como mostra a Figura 3.12, e clique em “OK” (Atividade 7.1).

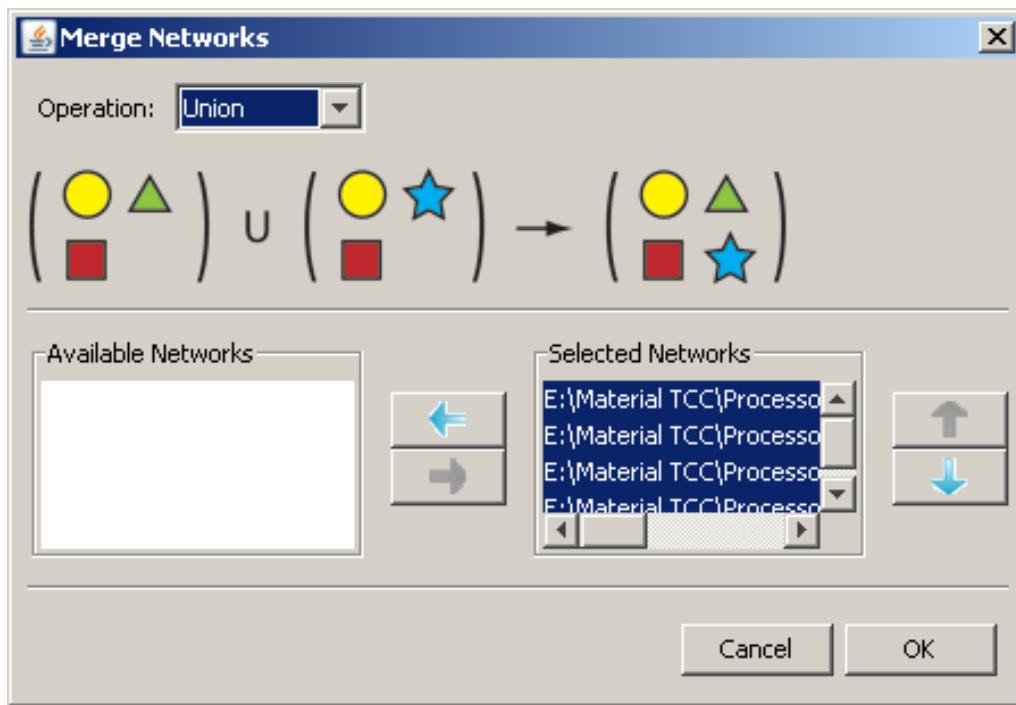


Figura 3.12: Merge das redes de interações das proteínas

Após isso você terá a representação gráfica do “merge” das redes de interações, como mostra a Figura 3.13 (Atividade 8.1).

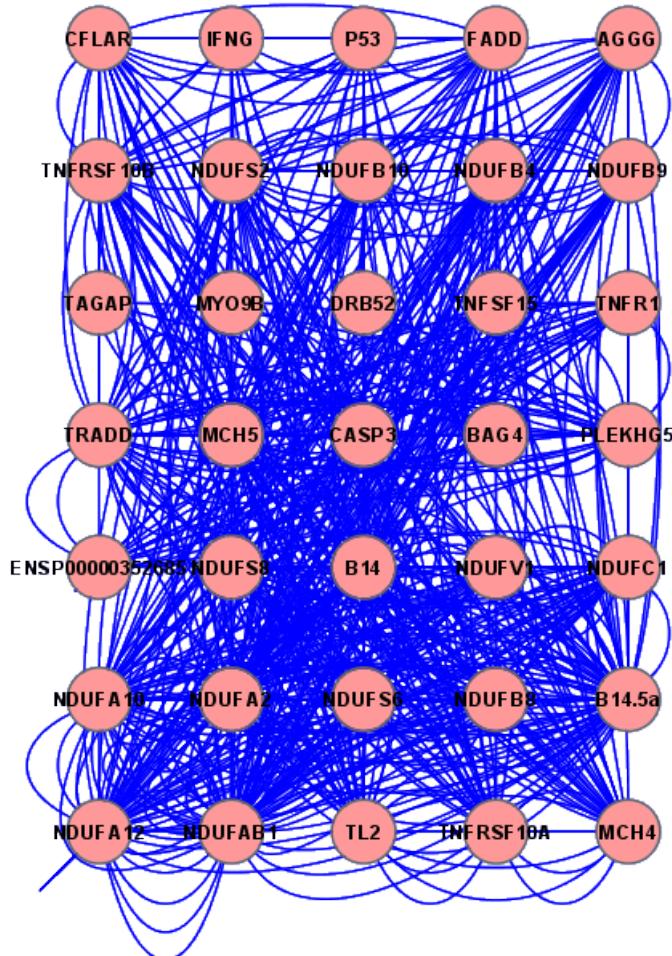


Figura 3.13: Representação gráfica das redes de interações

Uma vez que o especialista tenha essa rede de interações, ele pode fazer a análise dos agrupamentos dessa rede através do *plug-in* disponível para o software Cytoscape chamado “MCODE”. O MCODE² é responsável por encontrar *clusters* (regiões altamente conectadas) em uma rede, pois aglomerados significam coisas diferentes em tipos de redes diferentes (Atividade 9).

Então o especialista pode fazer a partição das redes de interação de proteínas usando a ontologia gênica através de outro *plug-in* disponível para o software Cytoscape chamado “BiNGO”. O BiNGO³ é responsável por determinar quais as categorias de ontologia gênica estão estatisticamente sendo representadas em um conjunto de genes ou o sub gráfico biológico de uma rede (Atividade 10).

²MCODE. Disponível em: <http://chianti.ucsd.edu/cyto_web/plugins/index.php>. Acesso em: 17 de junho de 2009

³BiNGO. Disponível em: <<http://www.psb.ugent.be/cbd/papers/BiNGO>>. Acesso em: 17 de junho de 2009

3.3.2 Descrição do Fluxo B

O Fluxo B de pesquisa ocorre da seguinte forma. Primeiramente acesse o *site* do OMIM, em <http://www.ncbi.nlm.nih.gov/omim>, e digite na caixa de texto ao lado do *label* “for” a doença que procura, como mostra a Figura 3.3 apresentada na Subseção 3.3.1, e depois clique no botão “Go” (Atividade 1).

Então o *site* lhe apresentará uma lista com as ocorrências da doença para que você selecione a que você está procurando, como mostra a Figura 3.4 apresentada na Subseção 3.3.1.

Após escolhida a ocorrência da doença, o *site* lhe apresentará um relatório com a descrição completa da mesma, como mostra a Figura 3.5 apresentada na Subseção 3.3.1 (Atividade 2).

Então localize e selecione uma ou mais proteínas/genes (no exemplo, sublinhadas em vermelho) no relatório, como mostra a Figura 3.6 apresentada na Subseção 3.3.1 (Atividade 3).

Acesse o *site* do STRING, em <http://string.embl.de>, e digite na caixa de texto localizada abaixo do *label* “list of names:” na aba “multiple names”, cada uma das proteínas selecionadas que deseja utilizar, como mostra a Figura 3.14 e clique no botão “GO !”.

The screenshot shows the STRING search interface. At the top, there are four tabs: "search by name", "search by protein sequence", "multiple names" (which is highlighted in blue), and "multiple sequences". Below the tabs is a large input field labeled "list of names:" with the instruction "(one per line; examples: #1 #2 #3)". Inside the field, the entries B8, DR3, B15, and DR4 are listed. Below this input field is a section labeled "... or upload a file:" with a file input field and a "Arquivo..." button. Further down is a "organism:" dropdown menu set to "auto_detect". At the bottom of the interface are several buttons: "COGs", a radio button labeled "Proteins" which is selected (indicated by a blue dot), "Reset", and a prominent blue "GO!" button.

Figura 3.14: Pesquisa das proteínas

O *site* lhe apresentará uma lista de organismos, como mostra a Figura 3.15, selecione o que deseja utilizar e clique no botão “Continue →” para prosseguir para a próxima etapa (Atividade 4.2).

Several organisms in STRING appear to match your input ...
Please select one from the list below, then click 'Continue' to proceed.

[← Back](#) [Continue →](#)

organism	nr of matched items
<input checked="" type="radio"/> Homo sapiens	4
<input type="radio"/> Arabidopsis thaliana	3
<input type="radio"/> Gallus gallus	3
<input type="radio"/> Pan troglodytes	3
<input type="radio"/> Oryzias latipes	3
<input type="radio"/> Mus musculus	2
<input type="radio"/> Saccharomyces cerevisiae	2

Figura 3.15: Lista de organismos que possuem as proteínas

Então o site lhe apresentará uma nova lista pendendo que você selecione dentre as ocorrências das proteínas quais serão utilizadas, como mostra a Figura 3.16, selecione-as e clique no botão “Continue →” novamente. Pode-se ressaltar que combinações diferentes geram redes diferentes (Atividade 5.2).

Então lhe será apresentada a rede de interação das proteínas, como mostra a Figura 3.17, clique no botão “save” para prosseguir (Atividade 6.2).

Após isso o site lhe apresentará uma última tela solicitando que você escolha o tipo de arquivo que deseja salvar, como mostra a Figura 3.18, selecione o arquivo do tipo XML (no exemplo, circulado em vermelho).

Uma vez que o arquivo esteja salvo em sua máquina local, abra o software “Cytoscape”, disponível em <http://www.cytoscape.org>, e abra o menu “File → Import → Network (multiple file types)...” (Atividade 7.2).

Selecione o arquivo que deseja importar clicando no botão “Select” e o localizando, então clique no botão “Import” para importar o arquivo, como mostra a Figura 3.11 apresentada na Subseção 3.3.1.

Após isso você terá a representação gráfica da rede de interação, como mostra a Figura 3.19 (Atividade 8.2).

Uma vez que o especialista tenha essa rede de interações, ele pode fazer a análise dos agrupamentos dessa rede através do *plug-in* disponível para o software Cytoscape chamado “MCODE” (Atividade 9).

Então o especialista pode fazer a partição das redes de interação de proteínas usando a ontologia gênica através de outro *plug-in* disponível para o software Cytoscape chamado “BiNGO” (Atividade 10).

The following proteins in *Homo sapiens* appear to match your input.
Please review the list, then click 'Continue' to proceed to the association network.

[← Back](#) [Continue →](#)

'B8':

NDUFA2 - subunit 2 (EC 1.6.5.3) (EC 1.6.99.3) (NADH-ubiquinone oxidoreductase B8 subunit) (Complex I-B8) (CI-B8)

HOXB8 - Homeobox protein Hox-B8 (Hox-2D) (Hox-2.4)

PI8 - Serpin B8 (Cytoplasmic antiproteinase 2) (CAP-2) (CAP2) (Protease inhibitor 8)

'DR3':

PLEKHG5 - DR3 (Apoptosis-mediating receptor TRAMP) (Death domain receptor 3) (WSL protein) (Apoptosis-inducing receptor AIR) (Apo-3) (Lymphocyte)

'B15':

NDUFB4 - subunit 4 (EC 1.6.5.3) (EC 1.6.99.3) (NADH-ubiquinone oxidoreductase B15 subunit) (Complex I-B15) (CI-B15)

ENSP00000341045 - UDP glycosyltransferase 2 family, polypeptide B15

'DR4':

ENSP00000353099 - HLA class II histocompatibility antigen, DRB1-1 beta chain precursor (MHC class I antigen DRB1*1) (DR-1) (DR1)

TNFRSF10A - Tumor necrosis factor receptor superfamily member 10A precursor (Death receptor 4) (TNF-related apoptosis-inducing ligand receptor 1) (TRAIL receptor 1) (TRAIL-R1) (CD261 antigen)

[← Back](#) [Continue →](#)

Figura 3.16: Lista de ocorrências das proteínas

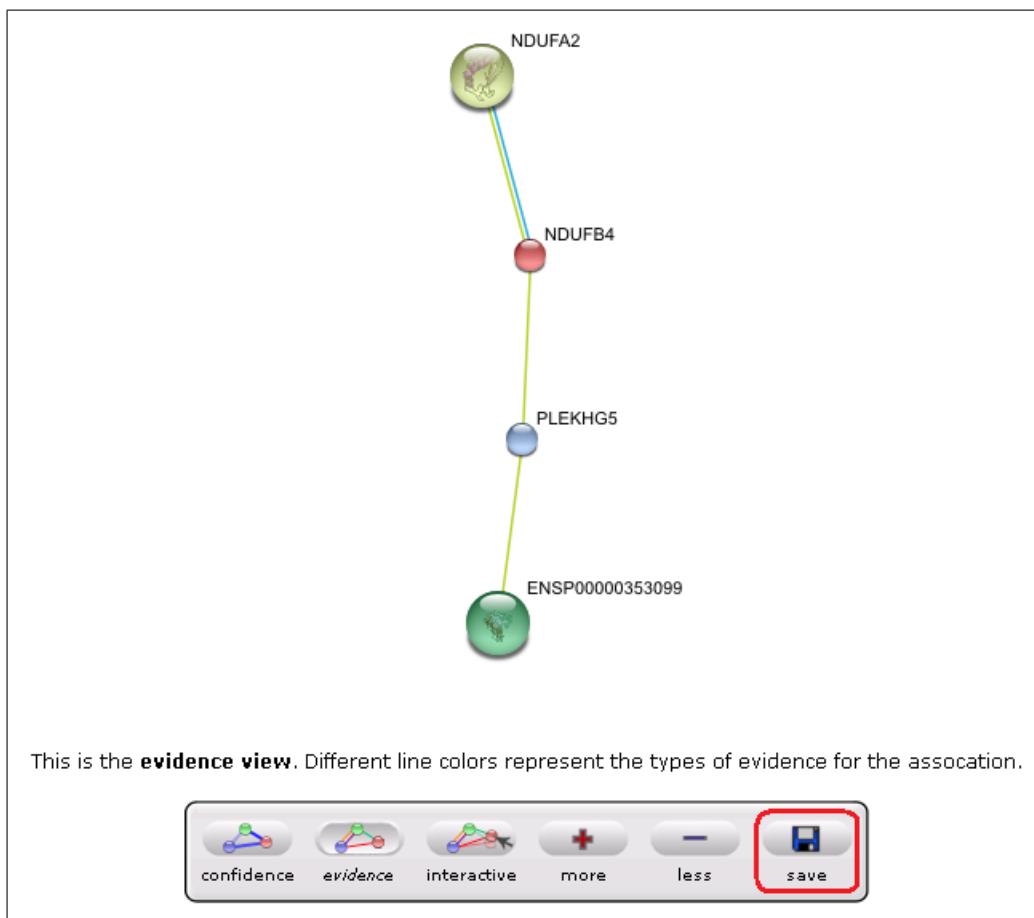


Figura 3.17: Apresentação da rede de interação das proteínas

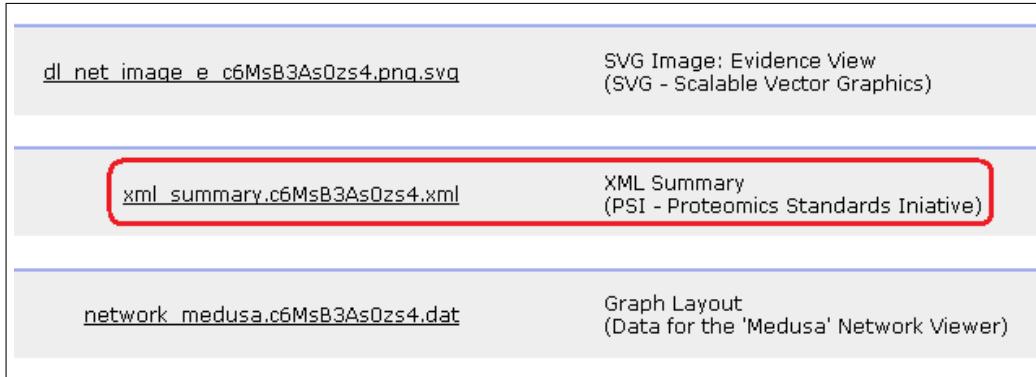


Figura 3.18: Seleção do tipo de arquivo das redes de interações

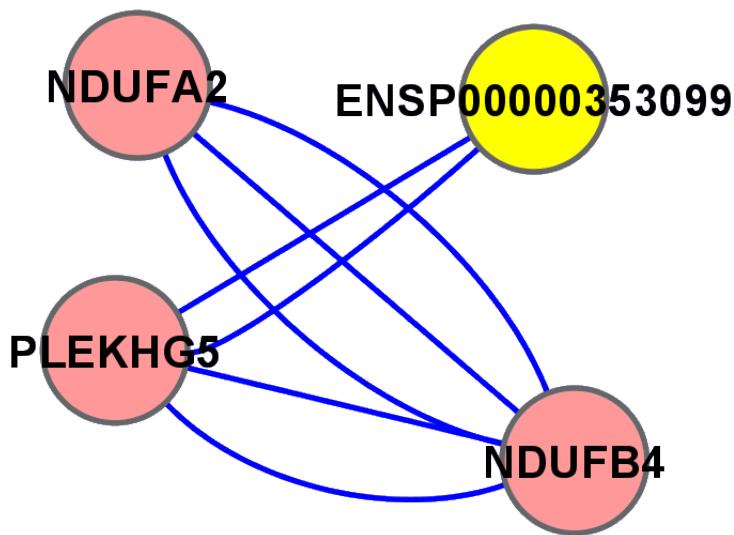


Figura 3.19: Representação gráfica das redes de interações

3.4 Considerações finais

Nesse capítulo foi apresentado o fluxo de pesquisa de uma doença gênica que será trabalhado nos próximos capítulos, explicando os conceitos necessários ao seu entendimento.

Para levantamento dos requisitos do sistema foi necessária uma reunião presencial com o especialista, nessa reunião foi explicado o fluxo de pesquisa de uma doença gênica, bem como os *sites* utilizados. Também foi levantada a forma que o fluxo era documentado para, posteriormente, repetir o experimento e descobriu-se que é feito de forma manual.

Então foram acessados os *sites* do OMIM e do STRING, pelo especialista, e foi mostrado passo-a-passo como eram feitas as buscas por doenças gênicas, por genes/proteínas no relatório da doença e, posteriormente, pelas redes de interação de proteínas. Outras dúvidas que surgiram durante a documentação do fluxo foram esclarecidas por *e-mail* com o especialista.

Feito isso, o fluxo de pesquisa de uma doença gênica foi documentado com a criação de um *workflow*, o qual demonstra os passos realizados pelo especialista. Esse *workflow* teve todos os seus passos descritos e explicados com o auxílio de imagens retiradas dos *sites*.

No próximo capítulo será apresentada a proposta de software para facilitar e tornar mais confiável esse fluxo de pesquisa.

4 PROPOSTA DE SOFTWARE

Neste capítulo serão apresentadas algumas ferramentas para o desenvolvimento de *workflows* científicos, as regras de nomenclatura genética e a modelagem do software a ser desenvolvido para utilizar a biologia de sistemas, para análise de uma rede de interação de proteínas a partir da pesquisa sobre doenças genéticas.

4.1 Workflow

Um *workflow* é uma seqüência de atividades ao longo de um processo de negócio, completo ou apenas parte dele, onde documentos, informações ou tarefas são transmitidas de um participante a outro por ações, de acordo com regras procedimentais (MATTOS et al., 2008; SOMMERVILLE, 2007).

Workflows científicos são definidos como recursos para resolução de problemas científicos através de técnicas tradicionais de *workflows*, ou seja, as idéias de execução de um conjunto de tarefas em uma determinada seqüência foram aproveitadas na área científica para a realização de experimentos e estudos. *Workflows* científicos diferem de *workflows* de negócio em diversos aspectos, particularmente em bioinformática, são caracterizados pelo alto grau de intervenção humana durante a sua execução (DIGIAMIETRI, 2007; SILVA, 2006). Nas subseções que seguem, serão apresentadas algumas ferramentas para o desenvolvimento de *workflows* científicos.

4.1.1 VisTrails

O VisTrails¹ foi concebido para gerenciar a rápida evolução dos *workflows*. VisTrails simplifica a criação, execução e compartilhamento de visualizações complexas, minerações de dados ou outros dados de análise em larga escala. Ao gerir automaticamente os dados, metadados, e os dados de exploração de processos, VisTrails permite que os usuários se concentrem em tarefas complexas e desafiadoras e os libera de tarefas tediosas que consomem muito tempo, como as relacionadas a organização e manipulação de grandes volumes de dados.

¹VisTrails. Disponível em: <<http://www.vistrails.org>>. Acesso em: 10 de junho de 2009

VisTrails fornece uma infra-estrutura que pode ser combinada com a de sistemas de visualização e de *workflow* existentes. Embora o VisTrails tenha sido originalmente construído para atender as necessidades de aplicações científicas exploratórias, a infra-estrutura que oferece é muito geral. Isto tornou claro que o sistema foi desenvolvido para pessoas de diferentes domínios, tanto para indústrias como para universidades. VisTrails tem potencial para reduzir o tempo de introspecção em praticamente qualquer tarefa exploratória.

4.1.2 Kepler

Kepler² foi concebido para ajudar os cientistas, os analistas e os programadores a criar, executar e compartilhar modelos e análises sobre uma vasta gama de disciplinas científicas e de engenharia. Kepler pode operar sobre os dados armazenados em uma variedade de formatos, locais e através da Internet, e é um meio eficaz para a integração de díspares componentes de software, ou facilitar a distribuição e execução de modelos remotos. Usando a interface gráfica do usuário, basta selecionar os usuários e em seguida, conectar componentes analíticos pertinentes e fontes de dados para criar um “trabalho científico” (representação de passos necessários para gerar resultados). O software Kepler ajuda os usuários a compartilhar e reutilizar dados, fluxos de trabalho e componentes desenvolvidos pela comunidade científica para resolver necessidades comuns.

O software Kepler é desenvolvido e mantido pelo *the cross-project Kepler collaboration*, que é liderado por uma equipe constituída por vários das principais instituições que originaram o projeto: UC Davis, UC Santa Barbara e UC San Diego. Primeiramente responsável pela concretização dos objetivos do Projeto Kepler a longo prazo, essa equipe trabalha para garantir a viabilidade técnica e financeira do Kepler, tomando as decisões estratégicas em nome da comunidade de usuários do Kepler, bem como proporcionar um ponto de contato oficial e duradouro para representar os interesses do Projeto Kepler.

Kepler é uma aplicação baseada em Java e é matido os sistemas operacionais Windows, OSX e Linux. O Projeto Kepler apóia o desenvolvimento do Kepler como código aberto, bem como fornece materiais e mecanismos para aprendizagem de como usar o Kepler, o compartilhamento de experiências com outros desenvolvedores de *workflow*, relatando *bugs*, sugerindo melhorias, etc.

²Kepler. Disponível em: <<https://kepler-project.org>>. Acesso em: 10 de junho de 2009

4.1.3 Taverna

O Taverna Workbench³ é uma ferramenta de software livre para projetar e executar os fluxos de trabalho, criada pelo Projeto myGrid e financiada pelo OMII-UK.

Taverna permite que os usuários integrem diversas ferramentas, incluindo os *web services* de diferentes domínios, como a química, a música e as ciências sociais. Para bionformática fornece acesso aos *services* prestados pela *National Center for Biotechnology Information, The European Bioinformatics Institute, the DNA Data-bank of Japan (DDBJ), SoapLab, BioMOBY e EMBOSS*.

O Taverna Workbench fornece um ambiente desktop para criação de *workflow* científicos. O *myExperiment social site* suporta procura e partilha de fluxos de trabalho e tem um suporte especial para Taverna *workflow*. O Taverna workbench, myExperiment e respectivos componentes são desenvolvidos e mantidos pela equipe do myGrid em colaboração com a comunidade de fonte aberta.

O Taverna roda em qualquer versão Windows, Linux, OSX e outros sistemas UNIX recentes. Se o seu computador tem uma conexão de rede e consegue executar o Java 5, você não precisa de mais nada, pois não existem bases de dados e nem análise de instalar aplicações, já que todos estes são acessados através da rede.

4.1.4 Egene

O Egene⁴ é um sistema genérico, flexível e modular para construção de *workflow* de trabalho, permite que programas de terceiros sejam utilizados e integrados segundo as necessidades de diferentes projetos e sem qualquer programação ou experiência a ser exigida.

O Egene vem com Coed, uma ferramenta visual que facilita a construção e documentação. Egene é um software de código aberto e foi desenvolvido para rodar em sistemas operacionais Unix/Linux. O sistema Egene foi escrito em Perl.

4.2 Nomenclatura genética

As regras de nomenclatura genética aprovadas pelo *HUGO Gene Nomenclature Committee* (HGNC) informam que, os nomes dos genes devem ser breves, específicos e devem tentar trazer informações sobre sua função e relação com os outros genes da mesma família (WAIN et al., 2009, 2004, 2002).

Também informam que, a primeira letra do nome do gene deve ser a mesma do símbolo do gene, para facilitar a localização, os nomes dos genes devem ser descritos na ortografia americana, a especificidade dos tecidos e o peso molecular devem ser

³Taverna Workbench. Disponível em: <<http://taverna.sourceforge.net>>. Acesso em: 10 de junho de 2009

⁴Egene. Disponível em: <<http://www.coccidia.icb.usp.br/egene>>. Acesso em: 10 de junho de 2009

evitados, podendo estes ser usados de forma ilimitada na descrição, e os nomes não devem usar termos para definir relações familiares com outros genes (SPLENDORE, 2005; WAIN et al., 2009, 2004, 2002). Segundo (HGNC, 2009; WAIN et al., 2009, 2004, 2002), os nomes de genes devem seguir as seguintes regras:

- Começam com letra minúscula, a não ser que seja o nome de uma pessoa que descreva a doença;
- Modificadores descritivos devem seguir a parte principal do nome, separados por vírgulas;
- Caso exista um nome alternativo esse deve ser colocado entre parênteses; e
- Caso exista um nome de outras espécies esse deve ser colocado entre parentes e no final do nome.

Os genes, além do nome oficial, também possuem um símbolo usado para designá-los em bancos de dados e publicações. Os símbolos dos genes são caracterizados por letras maiúsculas ou uma combinação de letras maiúsculas e algarismos arábicos, com exceção dos símbolos C#orf# (WAIN et al., 2009, 2004, 2002). Segundo (EYRE et al., 2006; FUNDEL; ZIMMER, 2006), os símbolos de genes devem seguir as seguintes regras:

- Devem ser curtos, preferencialmente com menos de seis caracteres de comprimento;
- Devem iniciar com uma letra maiúscula e podem ser seguidos de outras letras maiúsculas ou algarismos arábicos;
- Não podem conter sobre ou subscritos, algarismos romanos, letras gregas, pontuação (com exceção do gene HLA), “G” para a identificação de gene, ou qualquer outra referência a espécie, por exemplo, “H / h” para humanos;
- Devem ser evitadas referências a especificidade dos tecidos, peso molecular e localização cromossômica; e
- Também devem ser evitadas algumas letras ou combinação de letras que são usadas como prefixo ou sufixo em um símbolo para dar um significado específico, devendo essas não ser utilizadas para outros fins.

Embora existam inúmeras exceções, os genes e as proteínas derivadas carregam o mesmo nome/símbolo sendo essa inclusive uma recomendação do HGNC. Quando o mesmo símbolo é usado para designar o gene e a proteína, a maneira de diferenciá-los é pelo uso de itálico (por exemplo, gene *GATA1*, proteína GATA-1) (SPLENDORE, 2005; FUNDEL; ZIMMER, 2006). Na Tabela 4.1 é apresentado um resumo das regras de grafia para os símbolos dos genes humanos em trabalhos científicos.

Regras para grafia de genes humanos em trabalhos científicos			
Regras		Exemplos	
		Certo	Errado
Genes humanos devem ser grafados sempre em maiúsculas e em itálico		<i>BRCA1</i>	<i>Brca1</i>
Nos símbolos não são permitidos hífens, sobre ou subscritos, algarismos romanos ou letras do alfabeto grego		<i>IGF2</i> <i>TGFB3</i>	<i>IGF-II</i> <i>TGFβ3</i>
Não se deve usar "g" (de gene) ou "h" (de humano) antes do símbolo		<i>MSH2</i>	<i>hMSH2</i>
Não se deve usar o prefixo "c-" para designar oncogenes celulares		<i>MYC</i> <i>HRAS</i>	<i>c-myc</i> <i>c-H-ras</i>

Tabela 4.1: Regras de Grafia

Fonte: (SPLENDORE, 2005)

O benefício do uso de nomes e símbolos consistentes para os genes deve fazer com que um número de revistas cada vez maior passe a exigir nas suas publicações o cumprimento dessas regras, a exemplo do que já fazem as principais revistas da área biomédica, como a *Nature*, *Nature Genetics*, *American Journal of Human Genetics*, *Genomics*, *Human Mutation*, *Genes Chromosomes Cancer* e *The Lancet*. Para saber qual símbolo aprovado de um determinado gene, deve-se consultar a página do HGNC/Genew, ou então, procurá-lo no OMIM (*Online Mendelian Inheritance in Man*) (SPLENDORE, 2005; FUNDEL; ZIMMER, 2006).

4.3 Modelagem do software

Nessa seção serão apresentados os diagramas desenvolvidos com o objetivo de modelar o sistema *web* e melhorar o entendimento quanto ao funcionamento do mesmo.

4.3.1 Modelo de negócio - caso de uso

O sistema tem o objetivo de facilitar o processo de análise das redes de interação protéica e possibilitar armazenar informações quanto à pesquisa durante o processo para posteriormente poderem ser usadas caso seja necessário reproduzir os passos da pesquisa, para isso o diagrama de caso de uso da Figura 4.1 visa demonstrar a interação dos usuários “Bioinformática” e “Biólogo” com o “Sistema Web” e do “Sistema Web” com as outras aplicações.

No caso de uso, os usuário interagem apenas com o “Sistema Web” e com o software de visualização e análise da rede de interação de proteínas, ficando os acessos a outras aplicações, responsáveis pela pesquisa da doença e da proteína, transparentes para o usuário.

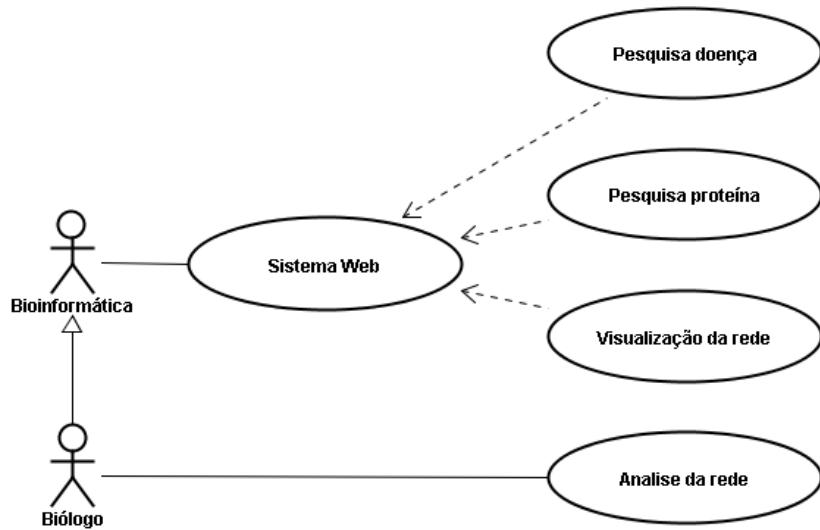


Figura 4.1: Caso de Uso

4.3.2 Workflow científico para análise de redes de interação protéica

O fluxo de pesquisa do software para geração da rede de interação da proteína foi otimizado para prever intervenção humana e será explicado a seguir, usando como base a Figura 4.2.

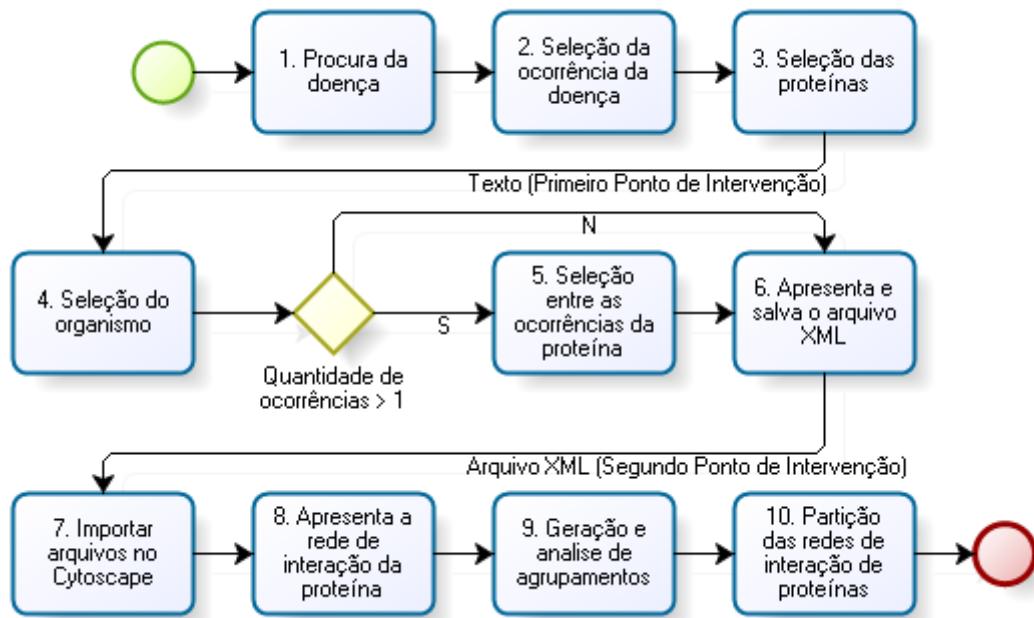


Figura 4.2: Fluxo de pesquisa do software

O fluxo se inicia com o usuário digitando o nome da doença que deseja encontrar (Atividade 1), o acesso ao site do OMIM é feito através da busca passando parâmetros pela url de chamada do site, por exemplo,

```
http : //www.ncbi.nlm.nih.gov/sites/entrez?db = omim&term = diabetes
```

onde o parâmetro “db” indica o banco de dados que está sendo utilizado e “term” a doença que está sendo procurada. Então o sistema apresenta as ocorrências encontradas para a doença e o usuário escolhe a que está procurando (Atividade 2). Após isso o sistema apresenta o relatório da doença e uma lista de sugestões de proteínas encontradas (Atividade 3). O sistema busca por essas proteínas no site do STRING passando parâmetros pela url de chamada do site, por exemplo,

```
http : //string.embl.de/newstring_cgi/show_network_section.pl?identifier = dr3
```

onde o parâmetro “identifier” indica a proteína que está sendo procurada. Então o sistema solicita que o usuário selecione o organismo que deseja utilizar (Atividade 4). Para o caso em que o site do STRING informar que foram encontradas mais de uma ocorrência para o nome de uma proteína, o sistema solicitará que o usuário escolha dentre as ocorrências quais serão utilizadas (Atividade 5). Após isso o sistema apresentará a rede de interação da(s) proteína(s) e retornará um arquivo XML (Atividade 6). Então esse arquivo XML será importado no software Cytoscape (Atividade 7) e será apresentada a representação gráfica da rede de interação da(s) proteína(s) (Atividade 8).

Uma vez que o especialista tenha essa rede de interações, ele pode fazer a análise dos agrupamentos dessa rede através do *plug-in* disponível para o software Cytoscape chamado “MCODE” (Atividade 9). Então o especialista pode fazer a partição das redes de interação de proteínas usando a ontologia gênica através de outro *plug-in* disponível para o software Cytoscape chamado “BiNGO” (Atividade 10).

4.3.3 Diagrama de arquitetura da aplicação

A arquitetura do sistema é enxuta, nela é necessária uma máquina servidora que tenha instalado um servidor *web* com a linguagem PHP e o SGBD (Sistema Gerenciador de Banco de Dados) MySQL.

Como pode ser acompanhado na Figura 4.3, o servidor recebe as requisições de serviços e retorna a página que estará disponível em um diretório virtual do servidor *web*. Essa página permitirá ao usuário interagir com as informações armazenadas no servidor, na base de dados MySQL e também com os outros serviços disponibilizados pelas páginas do OMIM e do STRING.

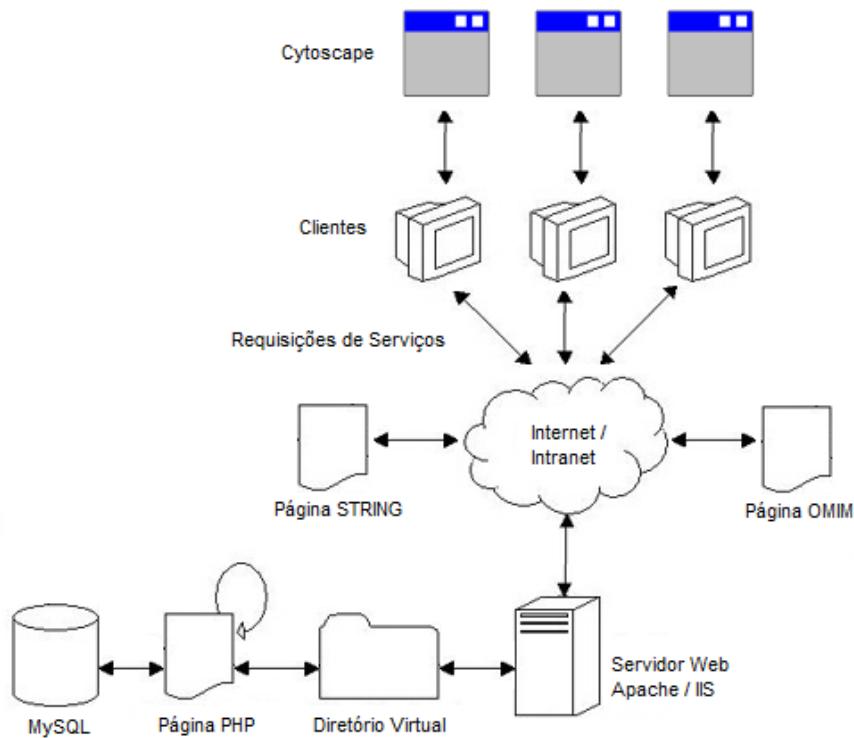


Figura 4.3: Diagrama de arquitetura da aplicação

4.4 Detalhamento da implementação

Nessa seção serão listadas e quando necessário detalhadas as funcionalidades que o sistema irá oferecer aos usuários, para tornar o processo ágil e seguro.

4.4.1 Algoritmo de extração dos dados

Expressões regulares podem ser definidas, “como um método formal de especificar um padrão de texto” (JARGAS, 2008). Porém se formos defini-las de uma forma mais detalhada, podemos dizer que descrevem uma linguagem exclusivamente através de símbolos e caracteres com funções especiais, que agrupados entre si e com caracteres literais formam uma seqüência, uma expressão, que entre outros pode vir a ser interpretada por uma linguagem de programação, ou um editor de textos (JARGAS, 2008; LEWIS; PAPADIMITRIOU, 2000).

Será desenvolvida uma expressão regular para ser aplicada no relatório da doença que é apresentado pelo *site* do OMIM. A expressão regular, como mostra a Figura 4.4, tem o objetivo de filtrar as palavras do texto que iniciem com letra maiúscula e que tenham apenas letras maiúsculas, números ou traço. Embora o exemplo tenha sido desenvolvido em Python a expressão pode ser usada praticamente em qualquer linguagem de programação sem sofrer alterações.

```

>>> texto = """
Aqui colocamos o texto que será utilizado na busca.
Podemos colocar alguns exemplos de formas erradas
para grafia de genes/proteínas, como por exemplo,
Brca1, hMSH2 e c-myc, que são encontradas em algumas
publicações. Também colocamos a forma certa para a
grafia de genes/proteínas, por exemplo, BRCA1, IGF2,
TGFB3, MYC, HRAS, B3 e HBA-1.
"""
>>> import re
>>> re.findall(r'\b[A-Z][A-Z0-9-]*\b', texto)
['BRCA1', 'IGF2', 'TGFB3', 'MYC', 'HRAS', 'B3', 'HBA-1']

```

Figura 4.4: Algoritmo de extração dos dados

4.4.2 Acesso aos sites do OMIM e do STRING

Primeiramente serão testadas outras formas de acesso aos *sites* do OMIM e do STRING, por exemplo, por *Web Service*. Conforme os testes que serão realizados com as formas de acesso, serão escolhidas as maneiras mais otimizadas de acessar os dados (Atividades 1 e 3).

4.4.3 Salvar e recuperar informações

O sistema permitirá salvar e recuperar consultas de doenças (Atividade 1), genes e proteínas escolhidas (Atividade 3), sendo que esses dois itens anteriores transmitem os dados em formato texto, redes de interação de proteínas (Atividade 6), sendo que esse é salvo em um arquivo do tipo XML, e a análise da ontologia gênica (Atividade 9), que é salva no formato do software Cytoscape.

O sistema permitirá salvar em banco de dados as consultas realizadas pelo especialista, os genes e proteínas já consultados anteriormente, *links* para redes de interação de proteínas já trabalhadas e informações do usuário para poder recuperar pesquisas personalizadas.

4.4.4 Interação com o software Cytoscape

A interação com o software Cytoscape se dará através da possibilidade de baixar o software através do sistema, para caso o especialista ainda não o tenha instalado em sua máquina, pesquisa por *plug-ins* que poderão vir a ser usados pelo especialista no software Cytoscape e a importação dos arquivos XML de redes de interação de proteínas apresentados pelo sistema a ser desenvolvido (Atividade 7).

4.5 Considerações finais

Nesse capítulo foram apresentados os artefatos desenvolvidos visando o desenvolvimento do sistema e algumas ferramentas existentes para a criação e execução de *workflows* científicos. Esse levantamento das ferramentas possibilitou o surgimento das idéias para melhorar o fluxo de pesquisa dos usuários que foram apresentadas.

Criou-se um *workflow* que visa demonstrar como será o fluxo de pesquisa no sistema, sendo esse explicado em detalhes, e foram desenvolvidos artefatos de software para demonstrar as interações dos usuários com o sistema. Também foi criado um esboço do algoritmo responsável por fazer a busca textual dos genes/proteínas no relatório da doença e foram levantados detalhes a serem melhor avaliados como o acesso aos *sites* do OMIM e do STRING, o armazenamento e manipulação dos dados e a integração com o software Cytoscape.

No próximo capítulo serão apresentados os artefatos e *scripts* desenvolvidos com base no sistema e um manual do usuário.

5 IMPLEMENTAÇÃO

Nesse capítulo serão apresentados o diagrama de arquitetura da aplicação, o diagrama de componentes, trechos dos *scripts* implementados e o *workflow* científico do sistema, sendo esses artefatos desenvolvidos com base em alterações no projeto original. Os *scripts* completos do sistema podem ser vistos no anexo A.

O sistema *web* BioNet facilita o processo dos usuários de bioinformática permitindo que sejam pesquisadas redes de interação de proteínas a partir de doenças gênicas, isso sem que seja necessário o uso direto das páginas *web* do OMIM e do STRING. O sistema se encarrega de fazer a comunicação com esses *sites* e ainda permite acompanhar todo o processo que está sendo realizado através de sua interface, podendo o usuário inclusive salvar e depois recuperar o processo executado.

5.1 Diagrama de arquitetura da aplicação

Como já havia sido descrito na subseção 4.3.3 do capítulo 4, a arquitetura do sistema continuou sendo enxuta, porém foi feita uma mudança na qual não é mais necessária a existência de um SGBD (Sistema Gerenciador de Banco de Dados). Como se constatou que os dados a serem armazenados são somente os passos executados, optou-se por permitir que o sistema salve e recupere esses dados a partir de um arquivo XML (*Extensible Markup Language*), podendo o especialista manipular esse arquivo dentro do sistema. A Figura 5.1 mostra o diagrama de arquitetura do sistema modificado.

5.2 Implementação do sistema web

O diagrama de componentes, mostrado na Figura 5.2, visa demonstrar como os componentes do sistema interagem entre si. Nas subseções que seguem será explicado o que faz cada um dos componentes, apresentando alguns trechos de código específicos responsáveis pelas funcionalidades do sistema. A implementação reflete os passos do *workflow* apresentado na subseção 4.3.2 do capítulo 4.

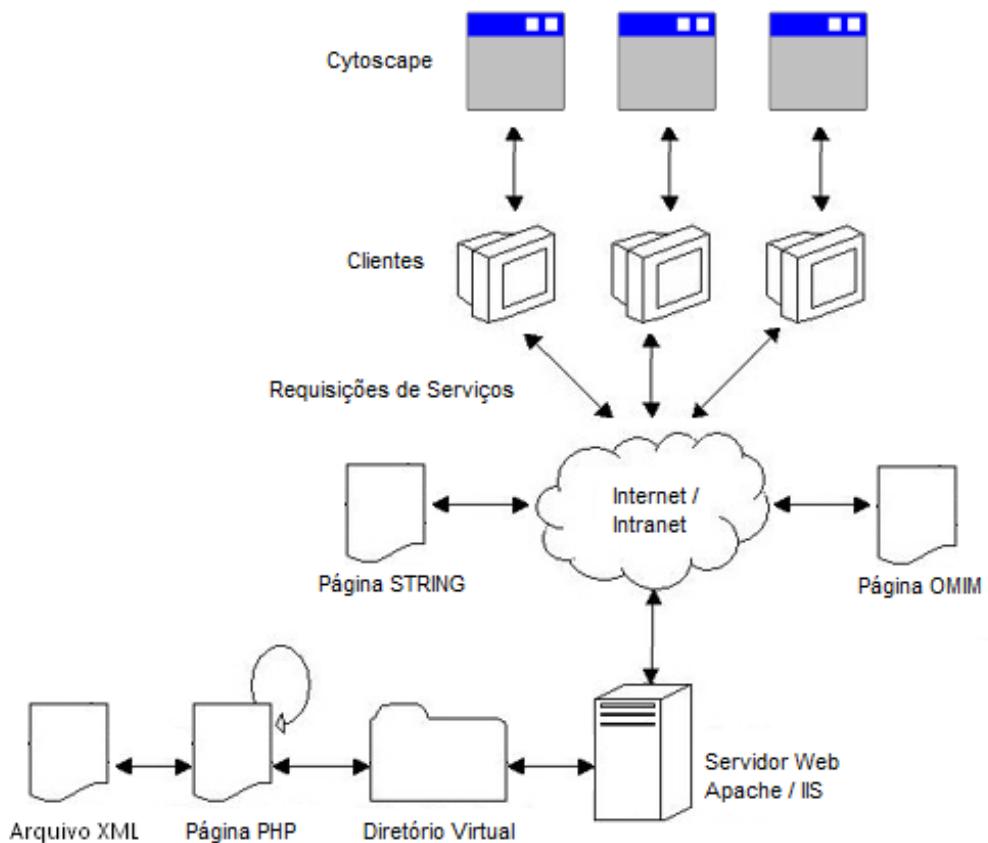


Figura 5.1: Diagrama de arquitetura da aplicação

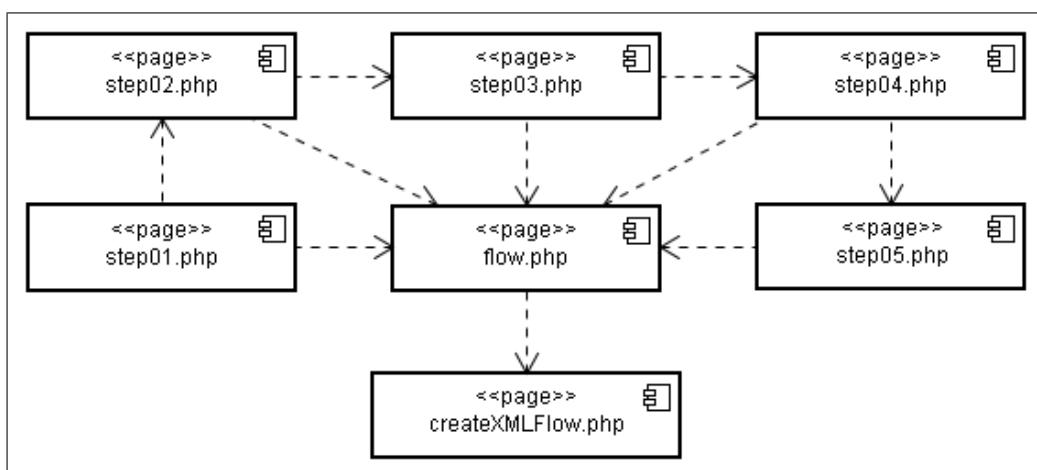


Figura 5.2: Diagrama de Componentes

5.2.1 Pesquisa da doença

O *script* (step01.php) do primeiro componente se resume a um formulário HTML (*HyperText Markup Language*) no qual o usuário digita o nome da doença que deseja encontrar e manda pesquisar.

5.2.2 Busca e seleção da doença

O *script* (step02.php) do segundo componente é responsável por fazer a busca e listar os resultados da pesquisa da doença, possibilitando ao usuário poder escolher a doença que está procurando, conforme projetado no caso de uso da Figura 4.1.

No *Script* 5.1 o termo pesquisado, anteriormente, é recebido (linha 4), então é informado o endereço do *Web Service* do site do OMIM (linhas 5 e 6) e depois ele é instanciado no sistema (linha 7). Após são capturadas as informações do *proxy* (linha 8), é verificado se não houve erros (linha 9), os parâmetros para a função que traz os identificadores das doenças com o termo são informados (linha 11) e então a função é chamada (linha 12). Tendo os identificadores armazenados em uma variável, é feito um *laço* (linha 13) no qual são buscadas algumas informações das doenças. Para isso, os parâmetros para a função que traz os nomes das doenças são informados (linha 14) e a função é chamada (linha 15), após isso temos algumas informações da doença para ajudar o usuário na escolha (linhas 16 a 20).

```

1 <?php
2 date_default_timezone_set('America/Sao_Paulo');
3 require_once('nusoap/lib/nusoap.php');
4 $term = $_POST['doenca'];
5 $wsdl = 'http://eutils.ncbi.nlm.nih.gov/entrez/eutils/' .
6     'soap/v2.0/eutils.wsdl';
7 $webService = new nusoap_client($wsdl, 'wsdl');
8 $proxy = $webService->getProxy();
9 $error = $webService->getError();
10 echo $error ? $error : '';
11 $params01 = array('db' => 'omim', 'term' => $term);
12 $resp01 = $proxy->run_eSearch($params01);
13 foreach ($resp01[IdList][Id] as $idItem){
14     $params02 = array('db' => 'omim', 'id' => $idItem);
15     $resp02 = $proxy->run_eSummary($params02);
16     echo $resp02[DocSum][Id];
17     echo $resp02[DocSum][Item][0][ItemContent];
18     echo $resp02[DocSum][Item][1][ItemContent];
19     echo $resp02[DocSum][Item][2][ItemContent];
20     echo $resp02[DocSum][Item][3][ItemContent];
21 }
22 ?>

```

Script 5.1: step02.php (trecho)

5.2.3 Visualização do relatório e seleção das proteínas

O *script* (step03.php) do terceiro componente é responsável por apresentar o relatório da doença selecionada anteriormente e sugerir proteínas encontradas, conforme projetado no caso de uso da Figura 4.1.

No *Script* 5.2 o identificador da doença selecionada, anteriormente, é recebido (linha 2) e então é trazido o arquivo HTML da página *web* do *site* do OMIM com o relatório da doença (linhas 3 e 4). Feito isso todo o conteúdo da página HTML, que vem na forma de um vetor, é organizado e colocado em uma variável (linhas 5 e 6). Após é criada uma variável com a expressão regular (linha 7), conforme projetado na subseção 4.4.1 do capítulo 4 e que agora aparece melhorada e otimizada para a linguagem PHP, que é responsável por separar do relatório da doença as proteínas, e então a busca é realizada (linha 8).

Então é feito um *laço* (linha 10) para testar todas as proteínas. No primeiro teste (linhas 11 a 14) é verificado se o termo está na lista de proteínas encontradas. No segundo teste (linhas 15 a 18) é verificado se o termo encontrado está na lista de termos a serem retirados (linha 9).

```

1 <?php
2 $id = $_GET['id'];
3 $page = file('http://www.ncbi.nlm.nih.gov/entrez/' .
4   'dispomim.cgi?cmd=entry&id=' . $id);
5 foreach($page as $line)
6   $texto .= $line;
7 $er = '(\b[A-Z]{1}[A-Z0-9]{1,5}\b)';
8 preg_match_all($er, $texto, $matches, PREG_SET_ORDER);
9 $listaRetirar = 'HTML HEAD TITLE OMIM HUMAN VIRUS TYPE';
10 foreach($matches as $proteina){
11   foreach(split(' ', $proteinasEncontradas) as $val){
12     if($val == $proteina[0])
13       $existe = true;
14   }
15   foreach(split(' ', $listaRetirar) as $val){
16     if($val == $proteina[0])
17       $existe = true;
18   }
19   if(!$existe)
20     $proteinasEncontradas .= $proteina[0] . ' ';
21   $existe = false;
22 }
23 echo $proteinasEncontradas;
24 echo $texto;
25 ?>
```

Script 5.2: step03.php (trecho)

No terceiro é verificado se o termo passou nos dois testes antes de ser adicionado a lista de proteínas encontradas (linhas 19 e 20). Por fim o *script* apresenta a lista de proteínas encontradas e o relatório da doença (linhas 23 e 24).

O segundo teste (linhas 15 a 18) foi necessário devido à função que executa a expressão regular (linha 7) não diferenciar termos em maiúsculas e minúsculas, fazendo com que sejam encontrados termos que não são proteínas, como por exemplo, algumas *tags* HTML e parte do relatório que não são proteínas (linha 9).

5.2.4 Seleção das ocorrências das proteínas

O *script* (step04.php) do quarto componente é responsável por apresentar as ocorrências das proteínas selecionadas, em humanos, conforme projetado no caso de uso da Figura 4.1.

No *Script* 5.3 as proteínas selecionadas são recebidas (linha 2) e organizadas em uma lista (linha 3). Então é feito um *laço* (linha 4) que pega cada uma das proteínas e busca o arquivo texto no *site* do STRING com as ocorrências da mesma em humanos (linhas 5 a 7) (*species=9606* na linha 7 significa humanos), então é realizado um teste (linha 8) para verificar se o arquivo de retorno é um vetor. Após isso é feito um *laço* (linha 9) que pega cada item e separa o identificador da proteína (linha 14) e a descrição (linha 15), através da expressão regular (linha 10) que foi executada (linhas 11 e 12) e é responsável por separar as informações que serão utilizadas.

```

1 <?php
2 $proteinas = $_POST['proteinas'];
3 $listaProteinas = split(' ', $proteinas);
4 foreach($listaProteinas as $proteina){
5     $proteinaFile = file('http://string-db.org/api/' .
6         'tsv-no-header/resolve?identifier=' . $proteina .
7         '&species=9606');
8     if(is_array($proteinaFile)){
9         foreach($proteinaFile as $vals){
10             $er = '(([A-Z0-9.]+)\s(.+?)ens)\s(.*)';
11             preg_match_all($er, $vals, $matches,
12                           PREG_SET_ORDER);
13             foreach($matches as $match){
14                 echo $match[1];
15                 echo $match[3];
16             }
17         }
18     }
19 }
20 ?>

```

Script 5.3: step04.php (trecho)

5.2.5 Visualização da rede de interação da(s) proteína(s)

O *script* (step05.php) do quinto componente é responsável por apresentar a rede de interação da(s) proteína(s) e disponibilizar para *download* seu arquivo XML, conforme projetado no caso de uso da Figura 4.1.

No *Script* 5.4 os identificadores das proteínas selecionadas na quarta etapa são recebidos e armazenados em uma variável (linha 2), então é realizado um teste para verificar se o usuário selecionou alguma proteína (linha 3), tendo selecionado alguma proteína é feito um teste (linha 4) que verifica se uma única proteína foi selecionada. Se apenas uma proteína tiver sido selecionada (linha 4) o sistema montará o *link*, para a busca do arquivo texto no STRING, que contém o endereço da imagem da rede de interação da proteína (linhas 5 a 7).

```

1 <?php
2 $proteinasSelecionadas = $_POST[ 'proteinasSelecionadas' ];
3 if( count($proteinasSelecionadas) > 0){
4     if( count($proteinasSelecionadas) == 1)
5         $linkFile = 'http://string-db.org/api/url/' .
6             'network?identifier=' .
7             $proteinasSelecionadas[0];
8     elseif( count($proteinasSelecionadas) > 1){
9         for($i=0;$i<count($proteinasSelecionadas);$i++){
10             if($i != (count($proteinasSelecionadas)-1))
11                 $listaProteinas .=
12                     $proteinasSelecionadas[$i] . '%0A';
13             else
14                 $listaProteinas .=
15                     $proteinasSelecionadas[$i];
16         }
17         $linkFile = 'http://string-db.org/api/url/' .
18             'networkList?identifiers=' . $listaProteinas .
19             '&limit=0';
20     }
21 } else
22     $linkFile = NULL;
23 if($linkFile != NULL){
24     $urlFile = file($linkFile);
25     $er = '(e\_\_([A-Za-z0-9]{2}[A-Za-z0-9\_]*)\.)';
26     preg_match_all($er, $urlFile[0], $matches,
27         PREG_SET_ORDER);
28     echo $urlFile[0];
29     echo $matches[0][1];
30 }
31 ?>
```

Script 5.4: step05.php (trecho)

Havendo mais de uma proteína selecionada (linha 8) será montado o *link* para a busca do arquivo texto no STRING, que contém o endereço da imagem da rede de interação das proteínas (linhas 9 a 19) colocando entre os identificadores das proteínas um caractere especial (linha 12). Então é feito um teste (linha 23) para verificar se o *link* foi montado e em caso afirmativo é feita a busca do arquivo (linha 24). Dentro desse arquivo é tirado o endereço da imagem da rede de interação da(s) proteína(s) (linha 28) e através da expressão regular (linha 25) que foi executada (linhas 26 e 27), é separado o identificador da rede no STRING (linha 29) que permite buscar o arquivo XML, abrir a página dos outros arquivos para *download* e da rede de interação da(s) proteína(s) na página *web* do STRING.

5.2.6 Documentação do fluxo de pesquisa da doença

O *script* (flow.php) do sexto componente é responsável por documentar o fluxo de pesquisa da doença e controlar as alterações nessa documentação.

O *Script* 5.5 inicia com a inicialização de uma sessão (linha 2), então é realizado um teste para verificar se a sessão *fluxo* já está registrada (linha 3), se não estiver então a sessão é registrada (linha 4), caso esteja a variável *fluxo* (do tipo vetor) recebe o conteúdo da sessão com o mesmo nome (linha 6).

No primeiro teste a variável *textoUser* (responsável por capturar a entrada de dados do usuário no fluxo) recebe o conteúdo de uma requisição identificada com o mesmo nome (linha 7), após isso é realizado um teste para verificar se a variável contém alguma informação (linha 8), caso contenha, esse valor é acrescentado ao final do vetor (linha 9). No segundo teste a variável *textoSoft* (responsável por capturar a entrada de dados do sistema no fluxo) recebe o conteúdo de uma requisição identificada com o mesmo nome (linha 10), após isso é realizado um teste para verificar se a variável contém alguma informação (linha 11), caso contenha, esse valor é acrescentado ao final do vetor (linha 12).

No terceiro teste a variável *flowXML* (responsável por captura a entrada de dados do usuário de um arquivo XML de com um fluxo) recebe o conteúdo de uma requisição identificada com o mesmo nome (linha 13), após é realizado um teste para verificar se a variável contém alguma informação (linha 14), caso contenha, são colocadas em variáveis todas as informações do arquivo (linhas 15 a 18), após são realizados outros teste para verificar se tamanho do arquivo é maior que zero, se o nome do arquivo contém algum texto e se o arquivo é do tipo XML (linhas 19 e 20), em caso afirmativo para todos os testes, o arquivo é renomeado e movido para junto dos *scripts* do sistema (linhas 21 e 22). Após isso é realizado mais um teste para verificar se o arquivo existe onde deveria estar (linha 23), caso exista o arquivo XML é aberto e carregado na variável *xml* (linhas 24 a 27) e então é feito um *laço* para ler todos os nodos do arquivo e colocar os textos no vetor (linhas 28 e 29).

```

1 <?php
2 session_start();
3 if (!session_is_registered('fluxo'))
4     session_register('fluxo');
5 else
6     $fluxo = $_SESSION['fluxo'];
7 $textoUser = $_GET['textoUser'];
8 if($textoUser)
9     $fluxo[count($fluxo)] = $textoUser;
10 $textoSoft = $_GET['textoSoft'];
11 if($textoSoft)
12     $fluxo[count($fluxo)] = $textoSoft;
13 $flowXML = $_FILES['flowXML'];
14 if($flowXML){
15     $nome = $flowXML['name'];
16     $tipo = $flowXML['type'];
17     $tamanho = $flowXML['size'];
18     $tmpNome = $flowXML['tmp_name'];
19     if($tamanho > 0 and strlen($nome) > 1
20         and $tipo == 'text/xml'){
21         $caminho = realpath('..') . '/' . $nome;
22         move_uploaded_file($tmpNome, $caminho);
23         if(file_exists($caminho)){
24             $xmlstr = file_get_contents($caminho);
25             $xmlDoc = new domDocument();
26             $xmlDoc->loadXML($xmlstr);
27             $xml = simplexml_import_dom($xmlDoc);
28             foreach($xml->step as $val)
29                 $fluxo[count($fluxo)] = (string) $val;
30         }
31     }
32 }
33 $removeItem = $_GET['$removeItem'];
34 if($removeItem > -1){
35     if(count($fluxo) != 1){
36         unset($fluxo[$removeItem]);
37         array_unshift($fluxo, array_shift($fluxo));
38     }
39     else
40         unset($fluxo);
41 }
42 $limpaFluxo = $_GET['$limpaFluxo'];
43 if($limpaFluxo == 'yes')
44     unset($fluxo);
45 $_SESSION['fluxo'] = $fluxo;
46 ?>
```

No quarto teste a variável *removeItem* (responsável por captura a indicação de uma remoção feita pelo usuário) recebe o conteúdo de uma requisição identificada com o mesmo nome (linha 33), após é realizado um teste para verificar se a variável contém um número válido do vetor (linha 34), caso contenha, é verificado se o vetor contém mais de um item (linha 35), se tiver, o item indicado pelo usuário é removido (linhas 36 e 37), caso não tenha, o vetor é todo apagado (linha 40).

No quinto teste a variável *limpaFluxo* (responsável por captura a indicação da remoção de todos os itens feita pelo usuário) recebe o conteúdo de uma requisição identificada com o mesmo nome (linha 42), após é realizado um teste para verificar se a variável contém a confirmação da remoção (linha 43), em caso afirmativo, todo o conteúdo do vetor é removido. Por fim, o vetor é gravado na variável de sessão com o mesmo nome (linha 45). Após o vetor ter sido gravado ele pode ser visualizado com o comando *print_r(\$fluxo)*.

5.2.7 Salvamento do fluxo de pesquisa da doença

O *script* (createXMLFlow.php) do sétimo componente é responsável por criar e disponibilizar para *download* o arquivo XML que contém o fluxo de pesquisa da(s) doença(s).

O *Script* 5.6 inicia da mesma forma que o *Script* 5.5 e das linhas 1 a 6 eles são idênticos, para tanto essas linhas não serão explicadas novamente. Após ter sido carregada ou criada a sessão, na variável *filename* é armazenado o nome que terá o arquivo a ser criado (linha 7), então é criada a variável que contém o arquivo XML (linhas 8 e 9), após é criado o nodo *flow* (linhas 10) e em seguida ele é adicionado ao arquivo (linha 11). Uma vez criado o nodo *flow*, é feito um *laço* (linha 12) para criar um nodo *step* para cada item do vetor contendo o valor desse item (linha 13) e depois adiciona-lo dentro do nodo *flow* (linha 14). Feito isso o arquivo XML é salvo (linha 16).

As linhas 17 a 27 tem a única função de fazer com que o sistema solicite um local para salvamento do arquivo na máquina do usuário.

5.2.8 Integração com o software Cytoscape

A integração do sistema com o software Cytoscape ocorre através do arquivo XML da rede de interação da(s) proteína(s) que é fornecido pelo quinto componente (step05.php) e que deve ser salvo para, posteriormente, ser importado pelo Cytoscape e utilizados os *plug-ins* desejados.

```

1 <?php
2 session_start();
3 if (!session_is_registered('fluxo'))
4     session_register('fluxo');
5 else
6     $fluxo = $_SESSION['fluxo'];
7 $filename = 'flow.xml';
8 $xmlDoc = new domDocument('1.0', 'utf-8');
9 $xmlDoc->formatOutput = true;
10 $flow = $xmlDoc->createElement('flow');
11 $flow = $xmlDoc->appendChild($flow);
12 for($i = 0; $i < count($fluxo); $i++){
13     $step = $xmlDoc->createElement('step', $fluxo[$i]);
14     $step = $flow->appendChild($step);
15 }
16 $xmlDoc->save($filename);
17 header('Content-Type: application/save');
18 header('Content-Length: ' . filesize($filename));
19 header('Content-Disposition: attachment; filename=' .
20         $filename . '');
21 header('Content-Transfer-Encoding: binary');
22 header('Expires: 0');
23 header('Pragma: no-cache');
24 $fp = fopen($filename, 'r');
25 fpassthru($fp);
26 fclose($fp);
27 exit;
28 ?>
```

Script 5.6: createXMLFlow.php (trecho)

5.3 Workflow científico do sistema web

O *workflow* científico da Figura 5.3 visa demonstrar as etapas que o especialista precisa realizar no sistema *web* para obter a(s) rede(s) de interação da(s) proteína(s).

O processo no sistema se inicia com a pesquisa da doença, como pode ser visto na Figura 5.4. O usuário digita a doença que deseja encontrar e clica no botão *Search*.

Então o sistema apresenta as ocorrências de doenças com aquele termo, assim como poderia ser feito no site do OMIM, para que o usuário escolha a doença que deseja visualizar e clica sobre seu *link*, como mostra a Figura 5.5.

Após isso, o sistema irá apresentar o relatório da doença escolhida anteriormente (o mesmo apresentado pelo OMIM) e também irá sugerir algumas proteínas encontradas no relatório para que o usuário selecione as que deseja pesquisar, como mostra a Figura 5.6.

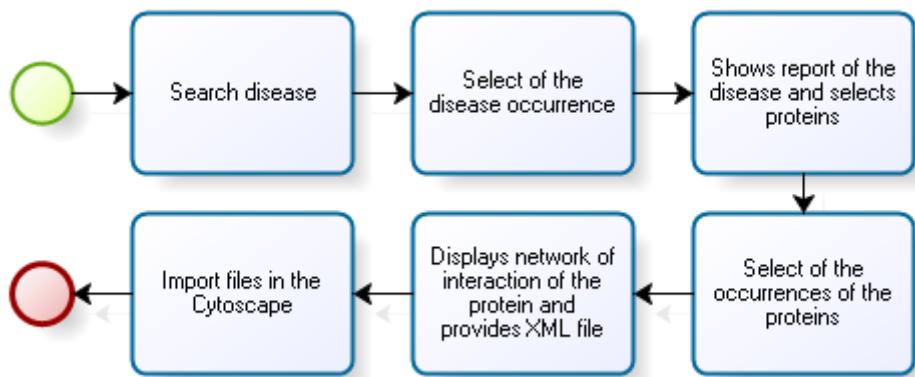


Figura 5.3: Fluxo Software

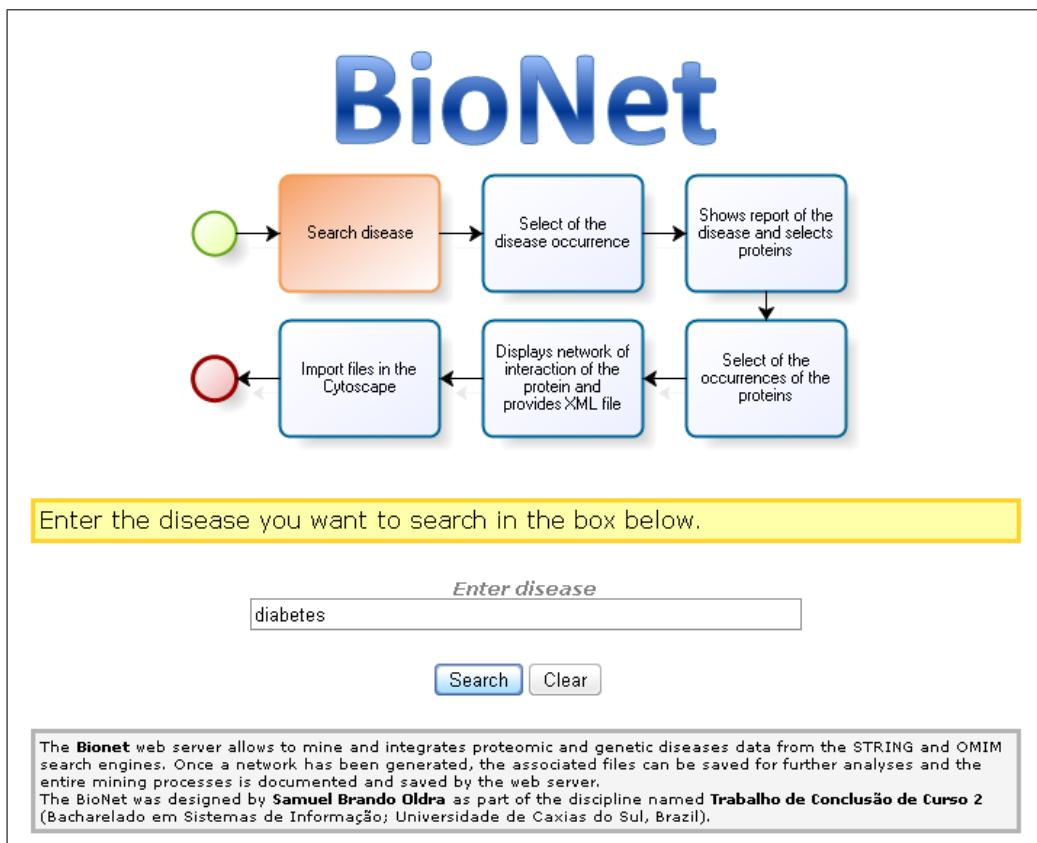


Figura 5.4: Procura da doença

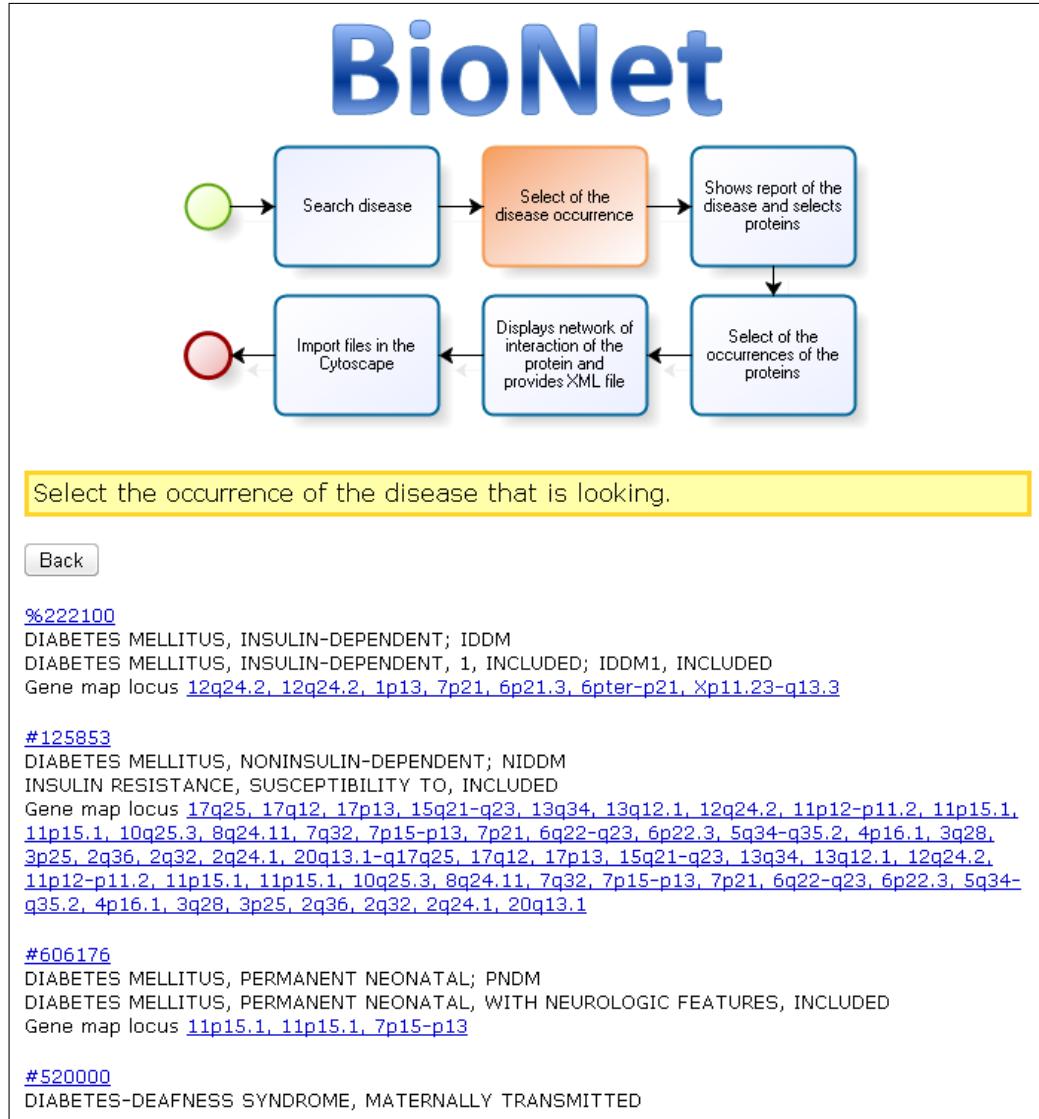


Figura 5.5: Seleciona a ocorrência da doença

Então o usuário pode apagar as proteínas que não deseja pesquisar da *caixa de texto* e, acrescentar as que deseja pesquisar como o sistema não encontrou ou não estão no relatório da doença, após isso o usuário clica no botão *Next* e o sistema irá pesquisar as ocorrências da(s) proteína(s).

Na seqüência, o sistema irá apresentar as ocorrências de cada proteína selecionada (em humanos), como mostra a Figura 5.7, assim como são apresentadas no site do STRING. O usuário então seleciona as que deseja visualizar a rede de interação da(s) proteína(s) e clica no botão *Next* novamente.

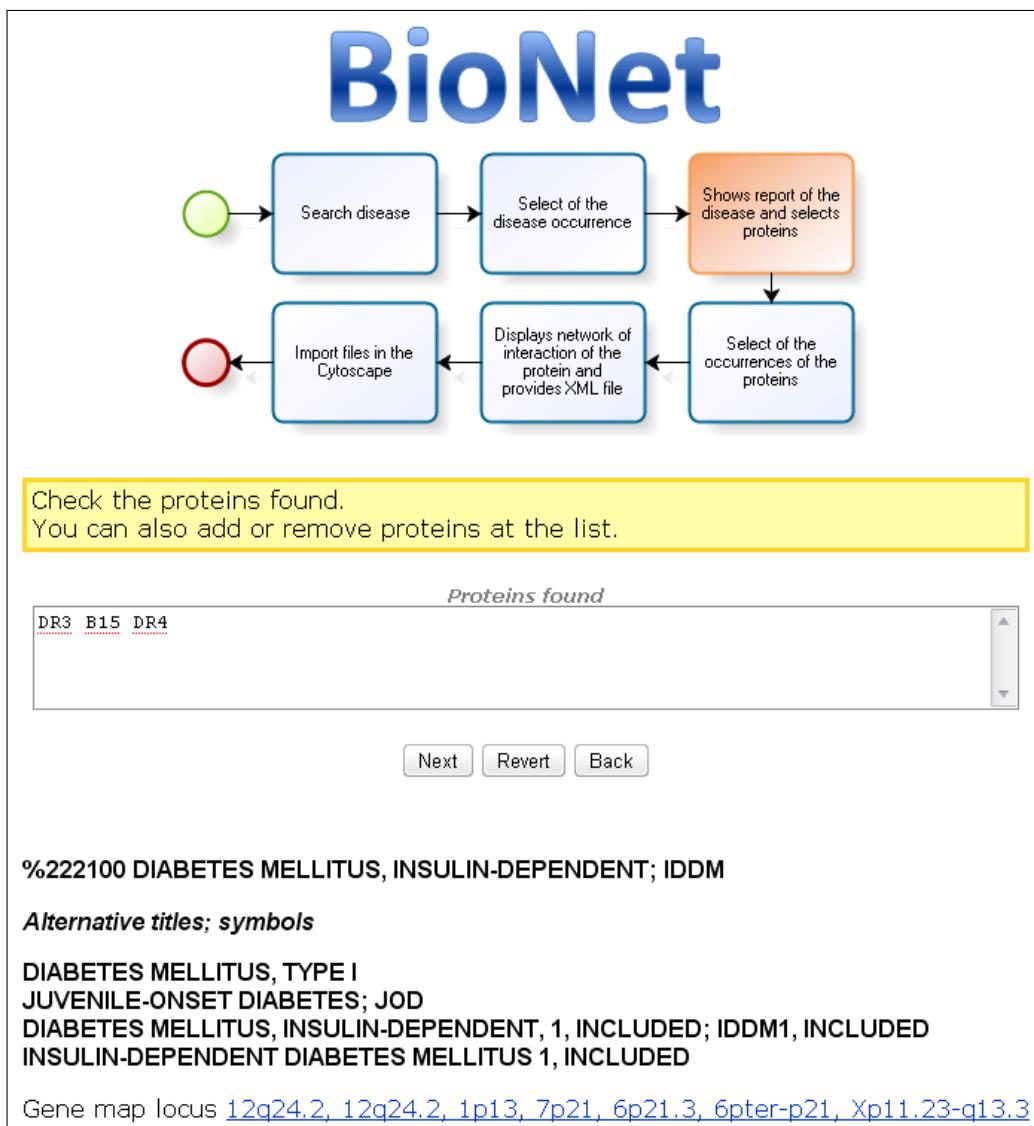


Figura 5.6: Apresenta relatório da doença e seleciona proteínas

Por fim, o sistema apresenta a imagem da rede de interação da(s) proteína(s), como pode ser visto na Figura 5.8, deixa disponível para *download* o arquivo XML clicando no link *Download XML*, que pode ser usado no software Cytoscape, possibilita a visualização dos outros arquivos que podem ser usados clicando no link *Other files* e clicando sobre a imagem o usuário será direcionado para a página do STRING para poder realizar qualquer alteração necessária.

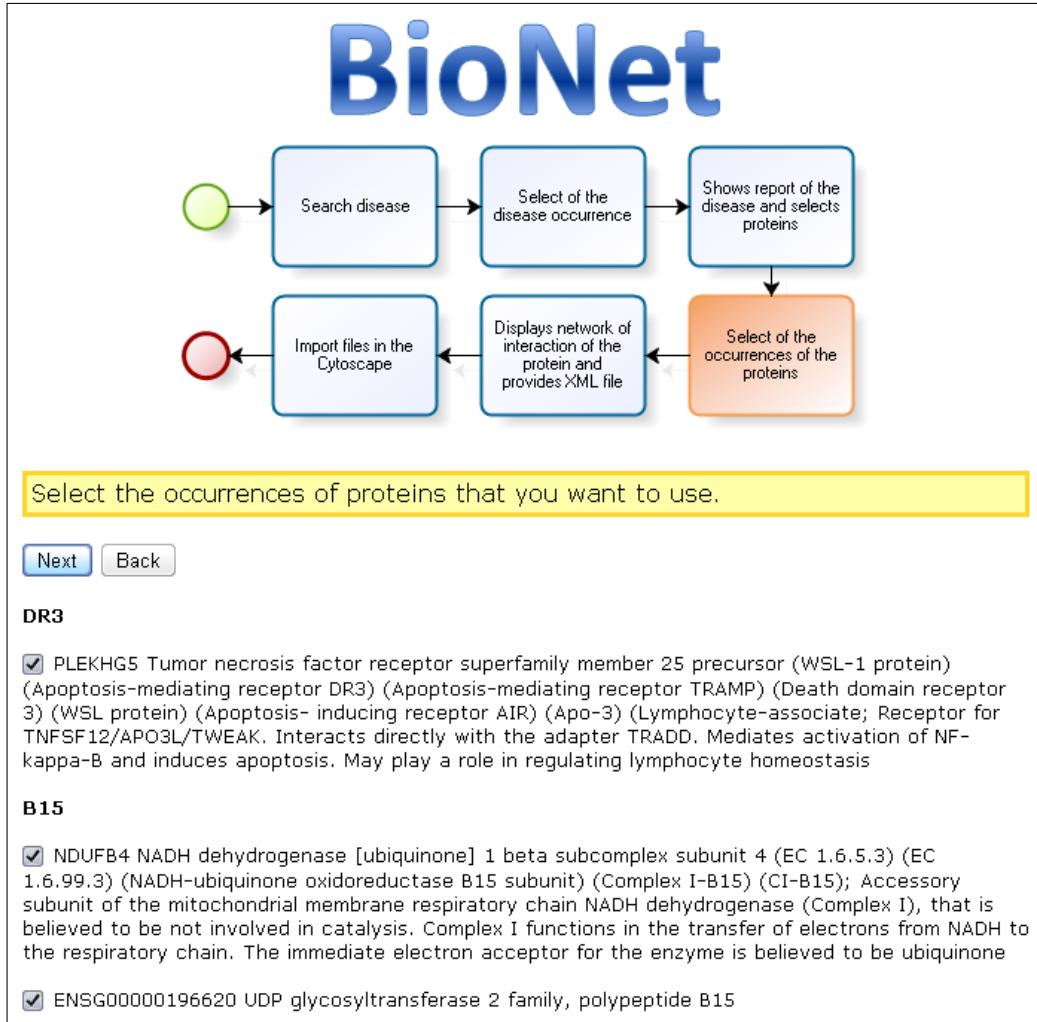


Figura 5.7: Seleciona as ocorrências das proteínas

Durante a execução do fluxo o usuário poderá observar que todos os passos estão sendo documentados, como mostra a Figura 5.9. Então o usuário pode fazer outras pesquisas na seqüência, limpar o fluxo clicando no link *Clean flow*, salvar o processo clicando com o botão direito do *mouse* no link *Download flow* e escolhendo um local para o arquivo, adicionar ao fim do fluxo um outro executado anteriormente e/ou recuperar um fluxo clicando no botão *Choose...*, localizando o arquivo XML do fluxo e após clicando no botão *Load*, adicionar comentários ao fluxo digitando a mensagem na caixa de texto *User comments* e após clicando no botão *Add* e apagar etapas ou comentários desnecessários no fluxo clicando sobre o link *remove* ao lado do mesmo.

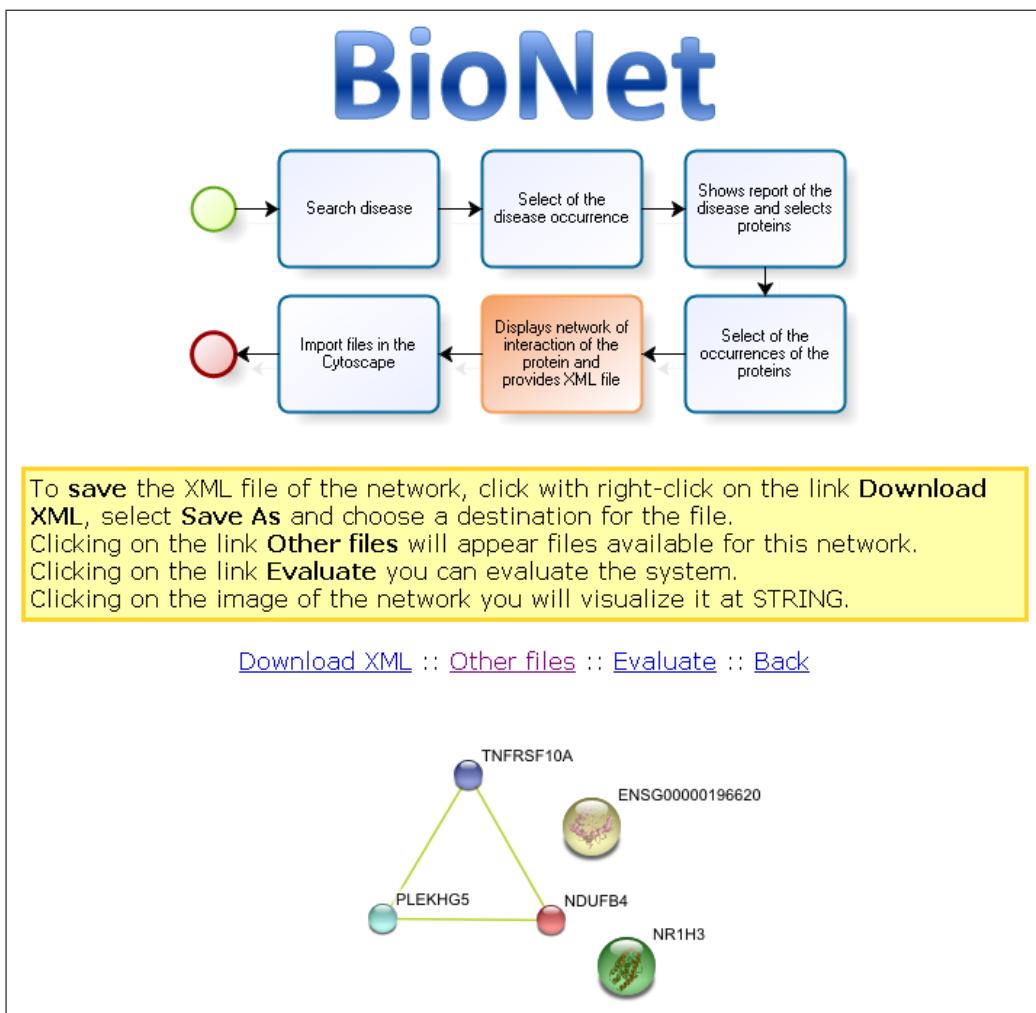


Figura 5.8: Apresenta rede de interação da proteína e fornece arquivo XML



Figura 5.9: Fluxo de pesquisa

5.4 Considerações finais

Nesse capítulo foram apresentados os artefatos e *scripts* desenvolvidos com base no sistema e um manual de funcionamento do sistema.

Durante o desenvolvimento do sistema, houve problemas com o sistema e mudanças na proposta de software, sendo essas explicadas a seguir. Optou-se por salvar os fluxos de pesquisa desenvolvidos pelo especialista em arquivos XML ao invés de em um SGBD (Sistema Gerenciador de Banco de Dados), visando deixar a cargo do especialista a manutenção de suas pesquisas, foram utilizadas formas de acesso diferentes das que haviam sido propostas, sendo usadas às indicadas pelos *sites* do OMIM e do STRING na maioria das ocasiões, o *workflow* do sistema foi modificado para deixar a escolha de uma ou mais proteínas transparente ao usuário, e existe um problema que não pode ser resolvido com relação à busca de proteínas não existentes no STRING. Também podemos mencionar que houve dificuldades com relação à hospedagem do sistema, sendo o atendimento e solicitações demorados.

No próximo capítulo serão apresentados os estudos de caso realizados com profissionais da área da biologia e da informática e suas opiniões com relação ao sistema.

6 ESTUDO DE CASO

Nesse capítulo serão apresentados os estudos de caso realizados com profissionais da área da informática e da biologia, a avaliação deles do sistema e minha avaliação como observador. O formulário de pesquisa pode ser visto no anexo B.

6.1 Primeiro estudo de caso

O primeiro estudo de caso foi realizado com o Prof. MSc. Daniel Luís Notari, profissional da área de informática, no dia 30 de novembro de 2009 às 17 horas e foi utilizado o navegador *web* “Firefox”.

No estudo, iniciou a pesquisa procurando pelo termo “alergics”, recebendo como resposta que o item não foi encontrado, na seqüência procurou pelos termos “sarampo”, “cachumba” e “caxumba” obtendo a mesma resposta, já que esses quatro termos não existem na base de dados do OMIM. Então procurou pelo termo “malaria”, obtendo assim a lista de ocorrências de doenças com esse termo. Selecionou a doença com o identificador “+109270” e com descrição principal “SOLUTE CARRIER FAMILY 4 (ANION EXCHANGER), MEMBER 1; SLC4A1”. Após isso foram apresentadas a descrição da doença e algumas sugestões de proteínas encontradas na mesma, então foram removidas algumas proteínas e termos que não eram proteínas, acrescentou-se a proteína “COX-1” e realizou a pesquisa pelas proteínas “SLC4A1”, “BND3”, “EMPB3”, “EPB3”, “AE1” e “COX-1”, obtendo assim a lista de ocorrências das proteínas em humanos. Então selecionou a única ocorrência da proteína “SLC4A1” e clicou para prosseguir, podendo assim visualizar a rede de interação da proteína, conforme mostra a Figura 6.1. Abaixo a descrição da proteína selecionada:

- *SLC4A1 Band 3 anion transport protein (Anion exchange protein 1) (AE 1) (Solute carrier family 4 member 1) (CD233 antigen); Band 3 is the major integral glycoprotein of the erythrocyte membrane. Band 3 has two functional domains. Its integral domain mediates a 1:1 exchange of inorganic anions across the membrane, whereas its cytoplasmic domain provides binding sites for cytoskeletal proteins, glycolytic enzymes, and hemoglobin.*

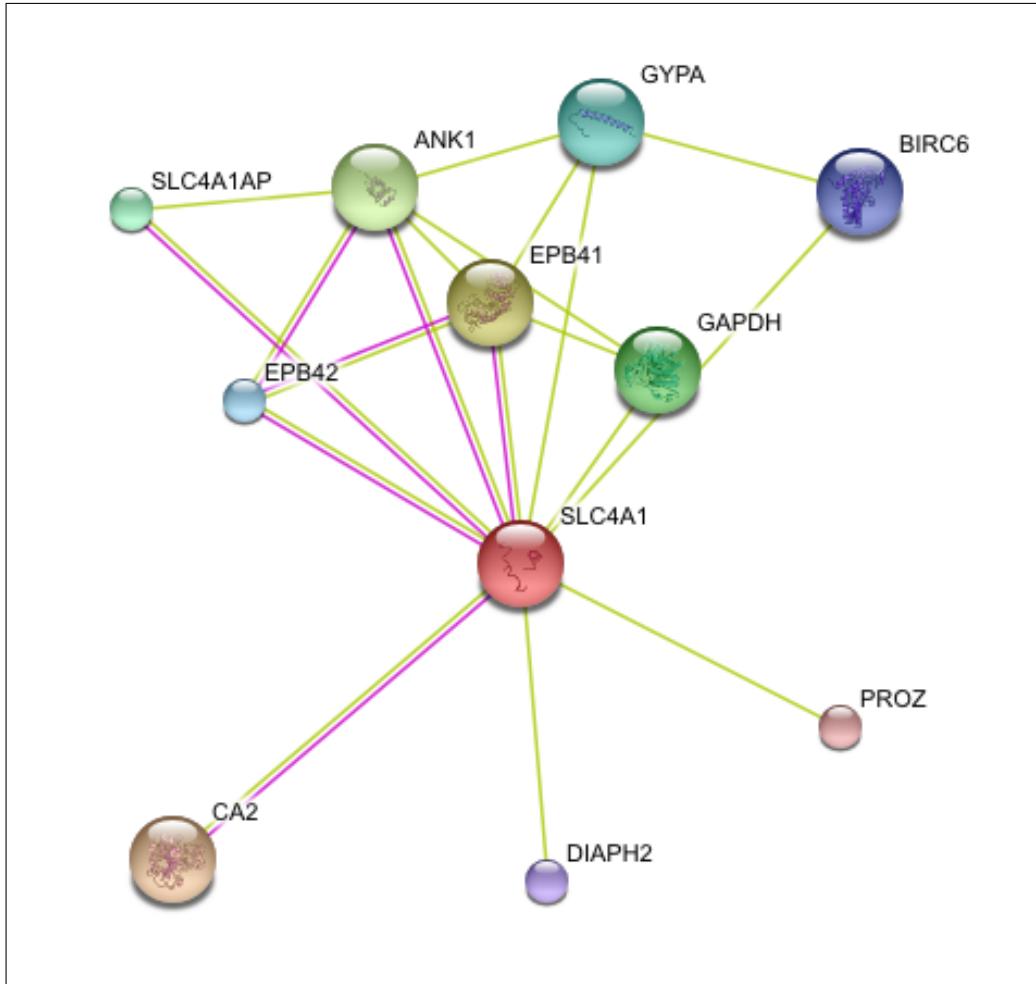


Figura 6.1: Primeiro estudo de caso rede STRING

Após isso realizou o *download* do arquivo XML da rede de interação da proteína, importou o mesmo no software Cytoscape, conforme mostra a Figura 6.2, e realizou alguns testes. Depois voltou ao sistema, realizou o *download* do arquivo XML do fluxo e respondeu ao questionário de avaliação do sistema.

Na avaliação realizada pelo Prof. MSc. Daniel Luís Notari o sistema foi considerado entre ótimo e bom, tendo o item com pior avaliação sido considerado regular, no caso, a busca de proteínas no relatório da doença.

6.2 Segundo estudo de caso

O segundo estudo de caso foi realizado com a Profa. Dra. Helena Graziottin Ribeiro, profissional da área de informática, no dia 1º de dezembro de 2009 às 17 horas e 30 minutos e foi utilizado o navegador web “Firefox”.

No estudo, iniciou a pesquisa procurando pelo termo “down syndrome”, obtendo assim a lista de ocorrências de doenças com esse termo. Selecionou a doença com o identificador “#190685” e com descrição principal “DOWN SYNDROME”. Após

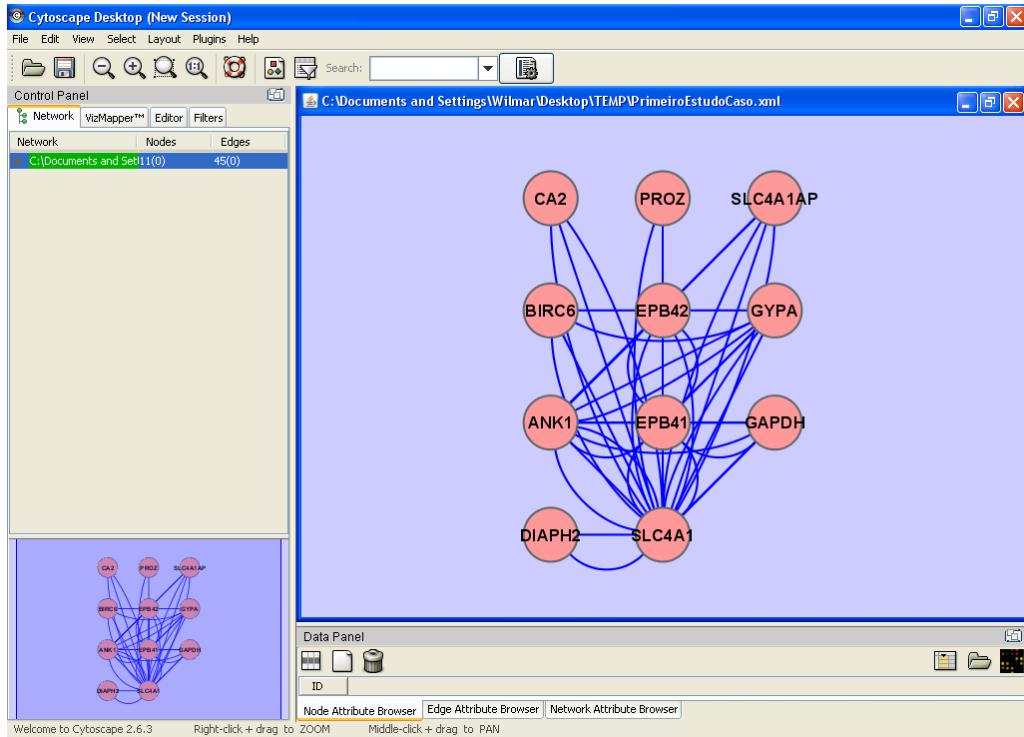


Figura 6.2: Primeiro estudo de caso rede Cytoscape

isso foram apresentadas a descrição da doença e algumas sugestões de proteínas encontradas na mesma, então foram removidas algumas proteínas e termos que não eram proteínas e realizou a pesquisa pelas proteínas “DOWN”, “REGION”, “DCR”, “DSCR”, “GATA1”, “ALL”, “AML”, “APP”, “DS”, “T4”, “CH” e “TSH”, obtendo assim a lista de ocorrências das proteínas em humanos. Durante essa pesquisa das ocorrências das proteínas, como nem todos os termos pesquisados eram proteínas o sistema apresentou “alertas” para os termos “DOWN” e “DSCR”. Então selecionou seis ocorrências da proteína “REGION” (que o princípio não é uma proteína) e clicou para prosseguir, podendo assim visualizar a rede de interação da proteína, conforme mostra a Figura 6.3. Abaixo as descrições das seis proteínas selecionadas:

- *DSCR10 Down syndrome critical region protein 10*;
- *CECR6 Cat eye syndrome critical region protein 6*;
- *DSCR4 Down syndrome critical region protein 4 (Down syndrome critical region protein B)*;
- *DSCR8 Down syndrome critical region protein 8 (Malignant melanoma-associated protein 1) (MMA-1) (MTAG2 protein)*;
- *DSCR6 Down syndrome critical region protein 6*;
- *DSCR9 Down syndrome critical region protein 9*.

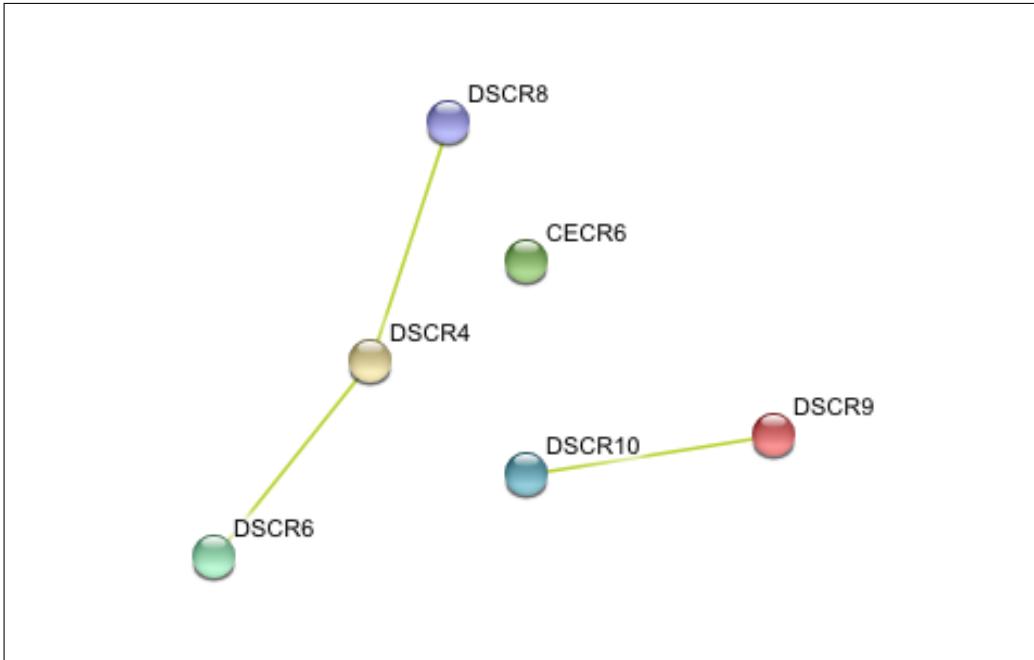


Figura 6.3: Segundo estudo de caso rede STRING

Após isso realizou o *download* do arquivo XML da rede de interação das proteínas, mas o mesmo não foi importado no Cytoscape por a Profa. não o ter instalado em sua máquina. Depois voltou ao sistema, realizou o *download* do arquivo XML do fluxo, limpou o fluxo, fez *upload* do arquivo XML do fluxo no sistema, inseriu um comentário no fluxo, depois o apagou e, por fim, respondeu ao questionário de avaliação do sistema.

Na avaliação realizada pela Profa. Dra. Helena Graziottin Ribeiro o sistema foi considerado bom, tendo o item com melhor avaliação sido considerado ótimo, no caso, a rede de interação da proteína apresentada como resultado. Nos comentário de sua avaliação também destacou que a visualização do fluxo durante toda a pesquisa ajuda a saber em que fase dela se está trabalhando, sendo isso de grande importância.

6.3 Terceiro estudo de caso

O terceiro estudo de caso foi realizado com a Profa. MSc. Scheila de Avila e Silva, profissional da área de biologia, no dia 2 de dezembro de 2009 às 16 horas e 30 minutos e foi utilizado o navegador *web* “Firefox”.

No estudo, iniciou a pesquisa procurando pelo termo “turner syndrome”, obtendo assim a lista de ocorrências de doenças com esse termo. Selecionou a doença com o identificador “#163950” e com descrição principal “NOONAN SYNDROME 1; NS1”. Após isso foram apresentadas a descrição da doença e algumas sugestões de proteínas encontradas na mesma, então foram removidos termos que não eram proteínas e realizou a pesquisa pelas proteínas “NS1”, “PTPN11”, “SHP2”, “SH2”,

“NF1”, “NFNS”, “KRAS” e “NS3”, obtendo assim a lista de ocorrências das proteínas em humanos. Durante essa pesquisa das ocorrências das proteínas, como um dos termos pesquisados não era uma proteína (“TPN11”) o sistema apresentou um “alerta” para o mesmo. Então selecionou quatro ocorrências de proteínas, duas da “NS1”, uma da “NF1” e uma da “KRAS”, e clicou para prosseguir, podendo assim visualizar a rede de interação da proteína, conforme mostra a Figura 6.4. Abaixo as descrições das quatro proteínas selecionadas:

- *CPSF4 Cleavage and polyadenylation specificity factor subunit 4 (Cleavage and polyadenylation specificity factor 30 kDa subunit) (CPSF 30 kDa subunit) (NS1 effector domain-binding protein 1) (Neb-1) (No arches homolog); Component of the cleavage and polyadenylation specificity factor (CPSF) complex that play a key role in pre-mRNA 3'-end formation, recognizing the AAUAAA signal sequence and interacting with poly(A) polymerase and other factors to bring about cleavage and poly(A) addition. CPSF4 binds RNA polymers with a preference for poly(U);*
- *PTPN11 Tyrosine-protein phosphatase non-receptor type 11 (EC 3.1.3.48) (Protein-tyrosine phosphatase 2C) (PTP-2C) (PTP-1D) (SH-PTP3) (SH-PTP2) (SHP-2) (Shp2); Acts downstream of various receptor and cytoplasmic protein tyrosine kinases to participate in the signal transduction from the cell surface to the nucleus;*
- *NF1 Neurofibromin (Neurofibromatosis-related protein NF-1) [Contains: Neurofibromin truncated]; Stimulates the GTPase activity of Ras. NF1 shows greater affinity for Ras GAP, but lower specific activity. May be a regulator of Ras activity;*
- *KRAS GTPase KRas precursor (K-Ras 2) (Ki-Ras) (c-K-ras) (c-Ki-ras); Ras proteins bind GDP/GTP and possess intrinsic GTPase activity.*

Após isso realizou o *download* do arquivo XML da rede de interação das proteínas, mas o mesmo não foi importado no Cytoscape por a Profa. não o ter instalado em sua máquina. Depois voltou ao sistema, realizou o *download* do arquivo XML do fluxo e respondeu ao questionário de avaliação do sistema.

Na avaliação realizada pela Profa. MSc. Scheila de Avila e Silva o sistema foi considerado ótimo, tendo o item com com pior avaliação sido considerado bom, no caso, a busca de proteínas no relatório da doença. Nos comentários de sua avaliação destacou que o sistema é útil para biólogos que trabalham com redes e relações de informações de diferentes bancos de dados, e que o trabalho deveria ser ampliado para integração de informações de outros bancos de dados.

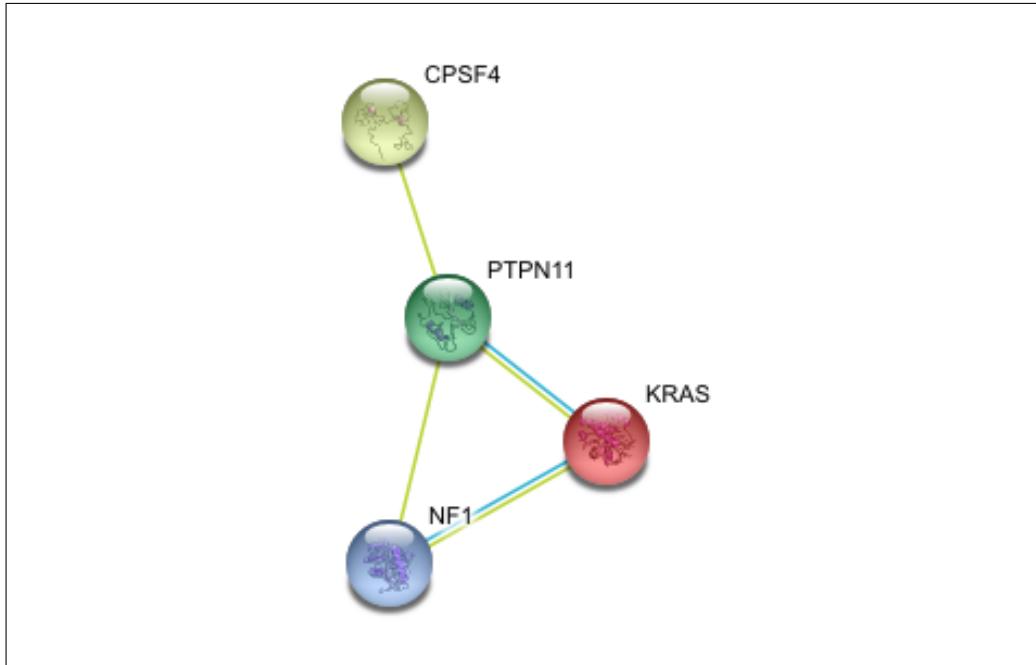


Figura 6.4: Terceiro estudo de caso rede STRING

6.4 Quarto estudo de caso

O quarto estudo de caso foi realizado com o Prof. Dr. Diego Bonatto, profissional da área de biologia, no dia 3 de dezembro de 2009 às 14 horas e 30 minutos e foi utilizado o navegador *web* “Firefox”. Nas subseções que seguem serão apresentadas as três pesquisas realizadas.

6.4.1 Primeira pesquisa

No primeiro estudo, iniciou a pesquisa procurando pelo termo “huntington”, obtendo assim a lista de ocorrências de doenças com esse termo. Selecionou a doença com o identificador “#143100” e com descrição principal “HUNTINGTON DISEASE; HD”. Após isso foram apresentadas a descrição da doença e algumas sugestões de proteínas encontradas na mesma, então foram removidos os termos que não eram proteínas e realizou a pesquisa pelas proteínas “HD”, “HTT”, “MRI”, “XL”, “DM1”, “NF1”, “NF2”, “D4S10”, “SE”, “G8” e “MMSE”, obtendo assim a lista de ocorrências das proteínas em humanos. Durante essa pesquisa das ocorrências das proteínas, como nem todos os termos pesquisados eram proteínas o sistema apresentou “alertas” para os termos “D4S10” e “MMSE”. Então selecionou quatorze ocorrências de proteínas, quatro da “HD”, uma da “HTT”, uma da “MRI”, uma da “XL”, uma da “DM1”, duas da “NF1”, uma da “NF2”, duas da “SE” e um da “G8”, e clicou para prosseguir, podendo assim visualizar a rede de interação das proteínas, conforme mostra a Figura 6.5. Abaixo as descrições das quatorze proteínas selecionadas:

- *OPTN Optineurin (Optic neuropathy-inducing protein) (E3-14.7K-interacting protein) (FIP-2) (Huntingtin-interacting protein HYPPL) (NEMO-related protein) (Transcription factor IIIA-interacting protein) (TFIIIA- IntP)*; Plays a neuroprotective role in the eye and optic nerve. Probably part of the TNF-alpha signaling pathway that can shift the equilibrium toward induction of cell death. May act by regulating membrane trafficking and cellular morphogenesis via a complex that contains Rab8 and huntingtin (HD). May constitute a cellular target for adenovirus E3 14.7, an inhibitor of TNF-alpha func [...];
- *HDDC3 HD domain-containing protein 3*;
- *PTPN23 Tyrosine-protein phosphatase non-receptor type 23 (EC 3.1.3.48) (His-domain-containing protein tyrosine phosphatase) (HD-PTP) (Protein tyrosine phosphatase TD14) (PTP-TD14)*; May act as a negative regulator of Ras-mediated mitogenic activity;
- *FBXO16 Zinc finger protein 395 (Papillomavirus-binding factor) (Papillomavirus regulatory factor 1) (PRF-1) (Huntington disease gene regulatory region-binding protein 2) (HDBP-2) (HD gene regulatory region-binding protein 2) (HD-regulating factor 2) (HDRF-2)*; Probably recognizes and binds to some phosphorylated proteins and promotes their ubiquitination and degradation;
- *SLC6A4 Sodium-dependent serotonin transporter (5HT transporter) (5HTT)*; Terminates the action of serotonin by its high affinity sodium-dependent reuptake into presynaptic terminals;
- *C7orf49 Uncharacterized protein C7orf49*; May act as a regulator of proteasome (By similarity);
- *RTN4 Reticulon-4 (Neurite outgrowth inhibitor) (Nogo protein) (Foocen) (Neuroendocrine-specific protein) (NSP) (Neuroendocrine-specific protein C homolog) (RTN-x) (Reticulon-5)*; Potent neurite growth inhibitor in vitro and plays a role both in the restriction of axonal regeneration after injury and in structural plasticity in the CNS. Isoform 2 reduces the anti-apoptotic activity of Bcl-xL and Bcl-2. This is likely consecutive to their change in subcellular location, from the mitochondria to the endoplasmic reticulum, after binding and sequestration. Isoform 2 and isoform 3 inhibit BACE1 ac [...];
- *DMPK Myotonin-protein kinase (EC 2.7.11.1) (Myotonic dystrophy protein kinase) (MDPK) (DM-kinase) (DMK) (DMPK) (MT-PK)*; Critical to the modulation of cardiac contractility and to the maintenance of proper cardiac conduction activity. Phosphorylates phospholamban;

- *NFIC Nuclear factor 1 C-type (Nuclear factor 1/C) (NF1-C) (NFI-C) (NF-I/C) (CCAAT-box-binding transcription factor) (CTF) (TGGCA-binding protein); Recognizes and binds the palindromic sequence 5'-TTGGC_{NNNN}NGCCAA-3' present in viral and cellular promoters and in the origin of replication of adenovirus type 2. These proteins are individually capable of activating transcription and replication (By similarity);*
- *APOBEC1 C->U-editing enzyme APOBEC-1 (EC 3.5.4.-) (Apolipoprotein B mRNA- editing enzyme 1) (HEPR); Catalytic component of the apolipoprotein B mRNA editing enzyme complex which is responsible for the posttranscriptional editing of a CAA codon for Gln to a UAA codon for stop in the APOB mRNA. Also involved in CGA (Arg) to UGA (Stop) editing in the NF1 mRNA;*
- *NF2 Merlin (Moesin-ezrin-radixin-like protein) (Neurofibromin-2) (Schwan-nomin) (Schwannomericlin); Probably acts as a membrane stabilizing protein. May inhibit PI3 kinase by binding to AGAP2 and impairing its stimulating activity;*
- *EZH2 Enhancer of zeste homolog 2 (ENX-1); Polycomb group (PcG) protein. Catalytic subunit of the PRC2/EED-EZH2 complex, which methylates 'Lys-9' and 'Lys-27' of histone H3, leading to transcriptional repression of the affected target gene. Able to mono-, di- and trimethylate 'Lys-27' of histone H3 to form H3K27me1, H3K27me2 and H3K27me3, respectively. Compared to EZH2-containing complexes, it is more abundant in embryonic stem cells and plays a major role in forming H3K27me3, which is required for embryonic stem cell identity and proper differentiation. The PRC2/EED-EZH2 complex may also se [...];*
- *FUT1 Galactoside 2-alpha-L-fucosyltransferase 1 (EC 2.4.1.69)(GDP-L-fucose: beta-D-galactoside 2-alpha-L-fucosyltransferase 1) (Alpha(1,2)FT 1) (Fucosyl-transferase 1) (Blood group H alpha 2- fucosyltransferase); Creates a soluble precursor oligosaccharide FuC-alpha ((1,2)Galbeta-) called the H antigen which is an essential substrate for the final step in the soluble A and B antigen synthesis pathway. H and Se enzymes fucosylate the same acceptor substrates but exhibit different Km values;*
- *G8 Protein G8.*

Após isso realizou o *download* do arquivo XML da rede de interação da proteína, importou o mesmo no software Cytoscape, mas devido as proteínas da rede não terem nenhuma ligação o Cytoscape apresentou uma tela em branco.

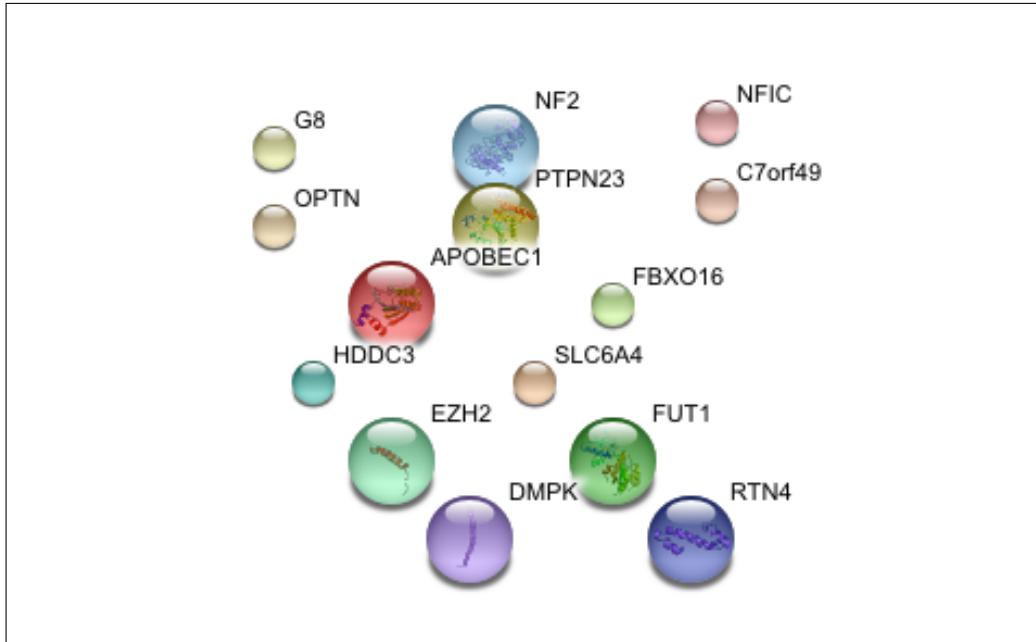


Figura 6.5: Quarto estudo de caso rede STRING primeira pesquisa

6.4.2 Segunda pesquisa

No segundo estudo, iniciou a pesquisa procurando pelo termo “charcot marie”, obtendo assim a lista de ocorrências de doenças com esse termo. Selecionou a doença com o identificador “#604563” e com descrição principal “CHARCOT-MARIE-TOOTH DISEASE, TYPE 4B2; CMT4B2”. Após isso foram apresentadas a descrição da doença e algumas sugestões de proteínas encontradas na mesma, então foram removidos os termos que não eram proteínas e realizou a pesquisa pelas proteínas “CMT4B2”, “SBF2”, “CMT4B1”, “MTMR2”, “CMT”, “CMT4A”, “CSF”, “CMT4B” e “MTMR13”, obtendo assim a lista de ocorrências das proteínas em humanos. Durante essa pesquisa das ocorrências das proteínas, como um dos termos pesquisados não era uma proteína (“CMT4A”) o sistema apresentou um “alerta” para o mesmo. Então selecionou quatorze ocorrências de proteínas, uma da “CMT4B2”, uma da “SBF2”, uma da “CMT4B1”, uma da “MTMR2”, uma da “CMT”, sete da “CSF”, uma da “CMT4B” e uma da “MTMR13”, e clicou para prosseguir, podendo assim visualizar a rede de interação da proteína, conforme mostra a Figura 6.6. Abaixo as descrições das proteínas selecionadas, sendo que as repetidas não serão citadas mais de uma vez:

- *SBF2 Myotubularin-related protein 13 (SET-binding factor 2); Not known;*
- *MTMR2 Myotubularin-related protein 2 (EC 3.1.3.-); Phosphatase that acts on lipids with a phosphoinositol headgroup. Has phosphatase activity towards phosphatidylinositol- 3-phosphate and phosphatidylinositol-3,5-bisphosphate;*
- *ENSG00000181464 CMT duplicated region transcript 1;*

- *CSF2RA Granulocyte-macrophage colony-stimulating factor receptor alpha chain precursor (GM-CSF-R-alpha) (GMR) (CD116 antigen) (CDw116); Low affinity receptor for granulocyte-macrophage colony- stimulating factor. Transduces a signal that results in the proliferation, differentiation, and functional activation of hematopoietic cells;*
- *NFATC2 Nuclear factor of activated T-cells, cytoplasmic 2 (T cell transcription factor NFAT1) (NFAT pre-existing subunit) (NF-ATp); Plays a role in the inducible expression of cytokine genes in T-cells, especially in the induction of the IL-2, IL-3, IL-4, TNF-alpha or GM-CSF;*
- *CSF3R Granulocyte colony-stimulating factor receptor precursor (G-CSF-R) (CD114 antigen); Receptor for granulocyte colony-stimulating factor (CSF3). In addition it may function in some adhesion or recognition events at the cell surface;*
- *CBFB Core-binding factor subunit beta (CBF-beta) (Polyomavirus enhancer-binding protein 2 beta subunit) (PEBP2-beta) (PEA2-beta) (SL3-3 enhancer factor 1 beta subunit) (SL3/AKV core-binding factor beta subunit); CBF binds to the core site, 5'-PYGPYGGT-3', of a number of enhancers and promoters, including murine leukemia virus, polyomavirus enhancer, T-cell receptor enhancers, LCK, IL-3 and GM-CSF promoters. CBFB enhances DNA binding by RUNX1;*
- *CSF2 Granulocyte-macrophage colony-stimulating factor precursor (GM-CSF) (Colony-stimulating factor) (CSF) (Sargramostim) (Molgramostin); Cytokine that stimulates the growth and differentiation of hematopoietic precursor cells from various lineages, including granulocytes, macrophages, eosinophils and erythrocytes;*
- *CSF1R Macrophage colony-stimulating factor 1 receptor precursor (EC 2.7.10.1) (CSF-1-R) (Fms proto-oncogene) (c-fms) (CD115 antigen); Protein tyrosine-kinase transmembrane receptor for CSF1 and IL34;*
- *IL10 Interleukin-10 precursor (IL-10) (Cytokine synthesis inhibitory factor) (CSIF); Inhibits the synthesis of a number of cytokines, including IFN-gamma, IL-2, IL-3, TNF and GM-CSF produced by activated macrophages and by helper T-cells.*

Após isso realizou o *download* do arquivo XML da rede de interação da proteína, importou o mesmo no software Cytoscape, conforme mostra a Figura 6.7, e realizou alguns testes. Depois voltou ao sistema e inseriu um comentário no fluxo antes de realizar o terceiro estudo de caso.

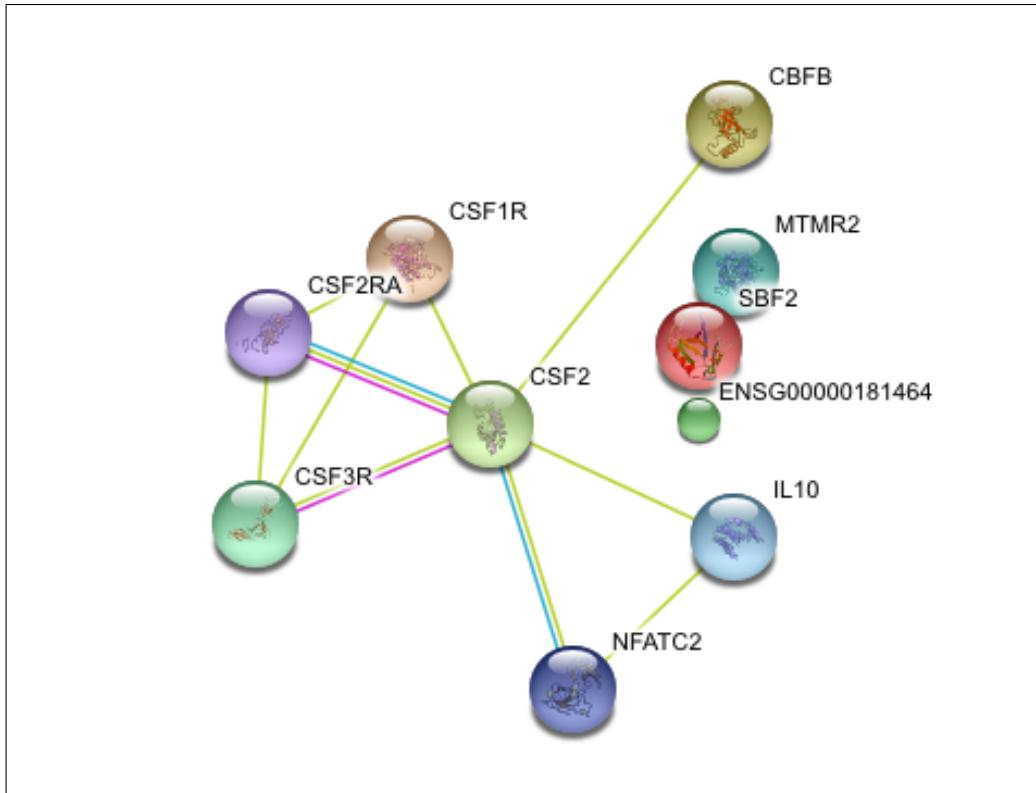


Figura 6.6: Quarto estudo de caso rede STRING segunda pesquisa

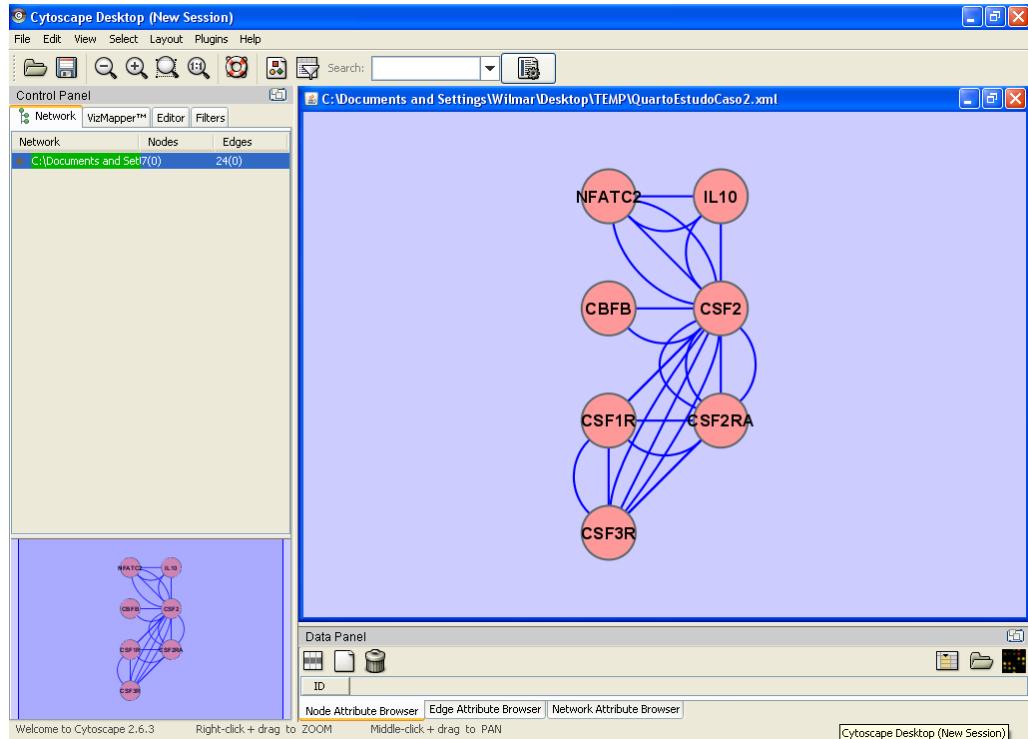


Figura 6.7: Quarto estudo de caso rede Cytoscape segunda pesquisa

6.4.3 Terceira pesquisa

No terceiro estudo, iniciou a pesquisa procurando pelo termo “cockayne”, obtendo assim a lista de ocorrências de doenças com esse termo. Selecionou a doença com o identificador “#216400” e com descrição principal “COCKAYNE SYNDROME, TYPE A; CSA”. Após isso foram apresentadas a descrição da doença e algumas sugestões de proteínas encontradas na mesma, então foram removidos os termos que não eram proteínas e realizou a pesquisa pelas proteínas “CSA”, “CKN1”, “ERCC8”, “CSB”, “ERCC6”, “ERCC3”, “ERCC2”, “ERCC5” e “COFS”, obtendo assim a lista de ocorrências das proteínas em humanos. Então selecionou treze ocorrências de proteínas, cinco da “CSA”, uma da “CKN1”, uma da “ERCC8”, uma da “CSB”, uma da “ERCC6”, uma da “ERCC3”, uma da “ERCC2”, uma da “ERCC5” e uma da “COFS”, e clicou para prosseguir, podendo assim visualizar a rede de interação da proteína, conforme mostra a Figura 6.8. Abaixo as descrições das proteínas selecionadas, sendo que as repetidas não serão citadas mais de uma vez:

- *ERCC8 DNA excision repair protein ERCC-8 (Cockayne syndrome WD repeat protein CSA); Involved in transcription;*
- *COPS3 COP9 signalosome complex subunit 3 (Signalosome subunit 3) (SGN3) (JAB1-containing signalosome subunit 3); Component of the COP9 signalosome complex (CSN), a complex involved in various cellular and developmental processes. The CSN complex is an essential regulator of the ubiquitin (Ubl) conjugation pathway by mediating the deneddylation of the cullin subunits of SCF-type E3 ligase complexes, leading to decrease the Ubl ligase activity of SCF-type complexes such as SCF, CSA or DDB2. The complex is also involved in phosphorylation of p53/TP53, c-jun/JUN, IkappaBalphal/NFKBIA, ITPK1 and I [...];*
- *COPS4 COP9 signalosome complex subunit 4 (Signalosome subunit 4) (SGN4) (JAB1-containing signalosome subunit 4); Component of the COP9 signalosome complex (CSN), a complex involved in various cellular and developmental processes. The CSN complex is an essential regulator of the ubiquitin (Ubl) conjugation pathway by mediating the deneddylation of the cullin subunits of SCF-type E3 ligase complexes, leading to decrease the Ubl ligase activity of SCF-type complexes such as SCF, CSA or DDB2. The complex is also involved in phosphorylation of p53/TP53, c-jun/JUN, IkappaBalphal/NFKBIA, ITPK1 and I [...];*
- *HSPA9 Stress-70 protein, mitochondrial precursor (75 kDa glucose-regulated protein) (GRP 75) (Peptide-binding protein 74) (PBP74) (Mortalin) (MOT);*

Implicated in the control of cell proliferation and cellular aging. May also act as a chaperone;

- *COPS6 COP9 signalosome complex subunit 6 (Signalosome subunit 6) (SGN6) (JAB1-containing signalosome subunit 6) (Vpr-interacting protein) (hVIP) (MOV34 homolog); Component of the COP9 signalosome complex (CSN), a complex involved in various cellular and developmental processes. The CSN complex is an essential regulator of the ubiquitin (Ubl) conjugation pathway by mediating the deneddylation of the cullin subunits of SCF-type E3 ligase complexes, leading to decrease the Ubl ligase activity of SCF-type complexes such as SCF, CSA or DDB2. The complex is also involved in phosphorylation of p53/ [...] ;*
- *ERCC6 DNA excision repair protein ERCC-6 (EC 3.6.1.-) (ATP-dependent helicase ERCC6) (Cockayne syndrome protein CSB); Is involved in the preferential repair of active genes. Presumed DNA or RNA unwinding function. Corrects the UV survival and RNA synthesis after UV exposure of Cockayne syndrome complementation group B;*
- *ERCC3 TFIIH basal transcription factor complex helicase XPB subunit (EC 3.6.1.-) (Basic transcription factor 2 89 kDa subunit) (BTF2-p89) (TFIIH 89 kDa subunit) (DNA-repair protein complementing XP-B cells) (Xeroderma pigmentosum group B-complementing protein); ATP-dependent 3'-5' DNA helicase, component of the core- TFIIH basal transcription factor, involved in nucleotide excision repair (NER) of DNA and, when complexed to CAK, in RNA transcription by RNA polymerase II. Acts by opening DNA either around the RNA transcription start site or the DNA damage;*
- *ERCC2 TFIIH basal transcription factor complex helicase subunit (EC 3.6.1.-) (DNA-repair protein complementing XP-D cells) (Xeroderma pigmentosum group D-complementing protein) (CXPD) (DNA excision repair protein ERCC-2); ATP-dependent 5'-3' DNA helicase, component of the core- TFIIH basal transcription factor. Involved in nucleotide excision repair (NER) of DNA by opening DNA around the damage, and in RNA transcription by RNA polymerase II by anchoring the CDK-activating kinase (CAK) complex, composed of CDK7, cyclin H and MAT1, to the core-TFIIH complex. Involved in the regulation of vitam [...] ;*
- *ERCC5 DNA-repair protein complementing XP-G cells (Xeroderma pigmentosum group G-complementing protein) (DNA excision repair protein ERCC-5); Single-stranded structure-specific DNA endonuclease involved in DNA excision*

repair. Makes the 3'incision in DNA nucleotide excision repair (NER). Acts as a cofactor for a DNA glycosylase that removes oxidized pyrimidines from DNA. May also be involved in transcription-coupled repair of this kind of damage, in transcription by RNA polymerase II, and perhaps in other processes too.

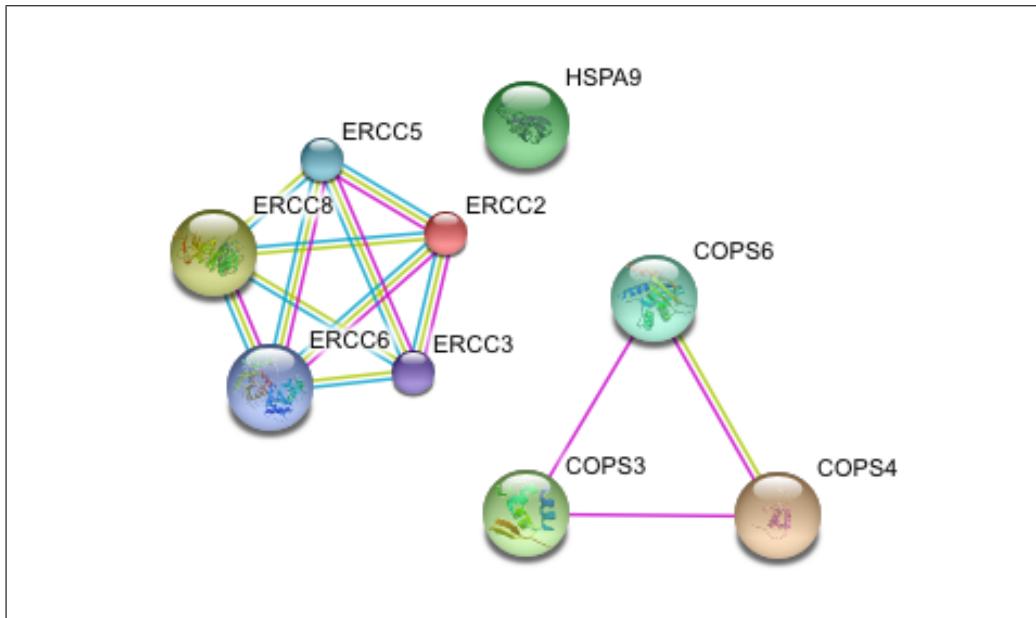


Figura 6.8: Quarto estudo de caso rede STRING terceira pesquisa

Após isso realizou o *download* do arquivo XML da rede de interação da proteína, importou o mesmo no software Cytoscape, conforme mostra a Figura 6.9, e realizou alguns testes. Depois voltou ao sistema, realizou o download do arquivo XML do fluxo, limpou o fluxo, fez *upload* do arquivo XML do fluxo no sistema e, por fim, respondeu ao questionário de avaliação do sistema.

Na avaliação realizada pelo Prof. Dr. Diego Bonatto o sistema foi considerado entre ótimo e bom. Nos comentários que fez do sistema destacou que o sistema será bastante útil e sugeriu que nas sugestões de proteínas encontradas no relatório da doença sejam mostradas mais proteínas, sendo umas trinta um bom número, e também que sejam desconsideradas proteínas com menos de três caracteres, pois geralmente não são usadas por ele e que se necessário podem ser encontradas aumentando o número de interações mínimas da rede de interação das proteínas no STRING.

6.5 Considerações finais

Na minha avaliação de observador percebi que as doenças foram pesquisadas, na maioria dos casos, com termos em português e não havia nada no sistema que especificasse que o termo deveria ser em inglês.

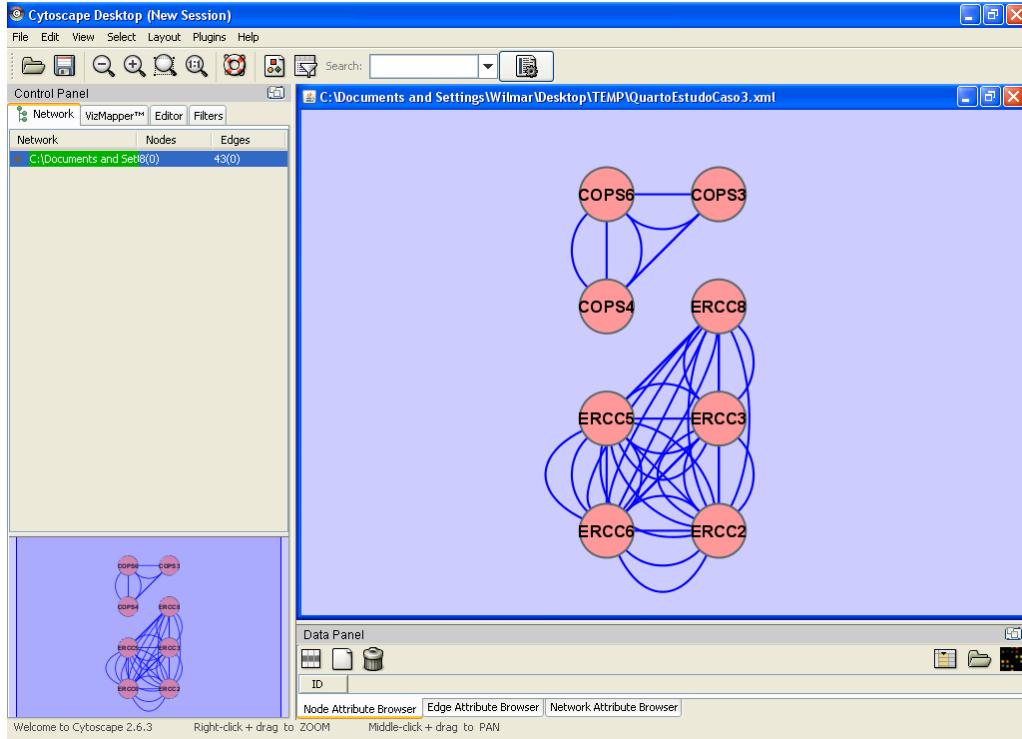


Figura 6.9: Quarto estudo de caso rede Cytoscape terceira pesquisa

E também que na busca de proteínas no relatório da doença, às vezes são trazidos termos que não são proteínas e mesmo isso não sendo um problema, visto que a idéia daquela busca é trazer uma lista de sugestões de proteínas, a busca por esses termos causa “alertas” no sistema, que mesmo não interrompendo o fluxo de pesquisa são visualmente desagradáveis ao usuário.

7 CONCLUSÃO

Durante o desenvolvimento desse trabalho foram realizados uma série de estudos sobre bioinformática, biologia molecular, bancos de dados biológicos, biologia de sistemas, ontologia gênica, entre outros, com o objetivo de entender os conceitos envolvidos e o fluxo de pesquisa de uma doença genética, executado pelo especialista e/ou biólogo. Também foram desenvolvidos artefatos visando à implementação do sistema e, após o sistema estar implementado, foram descritas as modificações no projeto, problemas enfrentados e o resultado final. Por fim, foram realizados quatro estudos de caso com mestres e doutores das áreas da biologia e da informática, no qual pode ser feita uma avaliação do sistema.

Como contribuição desse trabalho, foi desenvolvido um protótipo de sistema que automatiza e documenta o fluxo de pesquisa de uma doença gênica, integrando os dados dos *sites* do OMIM e do STRING e com isso simplificando o trabalho dos especialistas e/ou biólogos.

Mesmo o protótipo tendo alguns problemas, documentados nesse trabalho, o sistema teve uma boa avaliação e, principalmente, os especialista da área da biologia o consideraram útil e sugeriram uma série de possibilidades de continuação para esse trabalho.

Para trabalhos futuros, o sistema pode ser melhorado para atender melhor os seus usuários, por exemplo, adicionando funcionalidades como o refinamento das pesquisas com filtros por arquivos XML das redes, imagens das redes, entre outros, ou possibilitar o compartilhamento das pesquisas entre os usuários. E, visto que, esse trabalho mostrou aos biólogos que é possível integrar dados biológicos e automatizar fluxos de pesquisa, acredito que agora eles tenham inúmeras sugestões de trabalhos futuros interessantes.

O sistema *web* BioNet, desenvolvido nesse trabalho, está disponível para ser utilizado e testado em: <<http://www.biosoft.bio.br>>

REFERÊNCIAS

- BARABASI, A.-L.; OLTVAI, Z. N. Network biology: understanding the cell's functional organization. **Nature Reviews Genetics**, doi:10.1038/nrg1272, v.5, p.101–113, 2004.
- BEBEK, G.; YANG, J. Pathfinder: mining signal transduction pathway segments from protein-protein interaction networks. **BMC Bioinformatics**, doi:10.1186/1471-2105-8-335, v.8, n.335, 2007.
- BERG, J. M.; TYMOCZKO, J. L.; STRYER, L. **Bioquímica**. 5.ed. Rio de Janeiro: Guanabara Koogan, 2004.
- BHALLA, U. S. Understanding complex signaling networks through models and metaphors. **Progress in Biophysics and Molecular Biology**, doi:10.1016/S0079-6107(02)00046-9, v.81, p.45–65, 2003.
- CHAMPE, P. C.; HARVEY, R. A.; FERRIER, D. R. **Bioquímica ilustrada**. 3.ed. Porto Alegre: Artmed, 2006.
- DIGIAMIETRI, L. A. **Gerenciamento de workflows científicos em bioinformática**. 2007. Tese (Doutorado em Ciências da Computação) — UNICAMP (Campinas).
- EYRE, T. A.; DUCLUZEAU, F.; SNEDDON, T. P.; POVEY, S.; BRUFORD, E. A.; LUSH, M. J. The HUGO Gene Nomenclature Database, 2006 updates. **Nucleic Acids Research**, doi:10.1093/nar/gkj147, v.34, 2006.
- FERRO, M. **Desenvolvimento e validação de protocolos para a anotação automática de seqüências ORESTES de eimeria spp. de galinha doméstica**. 2008. Dissertação - Mestrado (Programa de Pós Graduação em Biologia da Relação Patógeno-Hospedeiro III) — Universidade de São Paulo.
- FUNDEL, K.; ZIMMER, R. Gene and protein nomenclature in public databases. **BMC Bioinformatics**, doi:10.1186/1471-2105-7-372, v.7, n.372, 2006.

- GIBAS, C.; JAMBECK, P. **Desenvolvendo bioinformática:** ferramentas de software para aplicação em biologia. Rio de Janeiro: Campus, 2001.
- HGNC. **HGNC Gene Families/Grouping Nomenclature.** Disponível em: <<http://www.genenames.org/genefamily.html>>. Acesso em: outubro de 2009.
- JARGAS, A. M. **Expressões regulares:** uma abordagem divertida. 2.ed. São Paulo: Novatec Editora, 2008.
- LESK, A. M. **Introdução à Bioinformática.** 2.ed. Porto Alegre: Artmed, 2008.
- LEWIN, B. **Genes VII.** Porto Alegre: Artmed, 2001.
- LEWIS, H. R.; PAPADIMITRIOU, C. H. **Elementos de teoria da computação.** 2.ed. Porto Alegre: Bookman, 2000.
- MATTOS, A.; SILVA, F. C. da; RUBERG, N.; CRUZ, S. M. S. da. **Gerência de Workflows Científicos:** Uma Análise Crítica no Contexto da Bioinformática. 2008. Dissertação - Mestrado (Programa de Engenharia de Sistemas de Computação) — Universidade Federal do Rio de Janeiro.
- MOTTA, V. T. da. **Bioquímica.** Caxias do Sul: Educs, 2005.
- NELSON, D. L.; COX, M. M. **Lehninger Princípios de Bioquímica.** 4.ed. São Paulo: Sarvier, 2006.
- O'MALLEY, M. A.; DUPRE, J. Fundamental issues in systems biology. **BioEssay**, v.27, n.12, p.1270–1276, 2005.
- PARIS, R. de. **Desenvolvimento de workflow científico para bioinformática.** 2008. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) — Universidade de Caxias do Sul.
- S. JUNIOR, C. da; SASSON, S. **Biologia.** 3.ed. São Paulo: Saraiva, 2003.
- SIEGAL, M. L.; PROMISLOW, D. E. L.; BERGMAN, A. Functional and evolutionary inference in gene networks: does topology matter? **Genetica**, doi:10.1007/s10709-006-0035-0, v.129, n.1, p.83–103, 2007.
- SILVA, F. N. **In Services:** Um sistema para gerenciamento de dados intermediários em workflows científicos na bioinformática. 2006. Dissertação (Mestrado em Sistemas e Computação) — Instituto Militar de Engenharia (Rio de Janeiro).
- SOMMERVILLE, I. **Engenharia de Software.** 8.ed. São Paulo: Pearson Addison-Wesley, 2007.

SPLENDORE, A. Para que existem as regras de nomenclatura genética? **Revista Brasileira de Hematologia e Hemoterapia**, doi:10.1590/S1516-84842005000200020, v.27, n.2, p.148–152, 2005.

WAIN, H. M.; BRUFORD, E. A.; LOVERING, R. C.; LUSH, M. J.; WRIGHT, M. W.; POVEY, S. Guidelines for Human Gene Nomenclature. **Genomics**, doi:10.1006/geno.2002.6748, v.79, n.4, p.464–470, 2002.

WAIN, H. M.; BRUFORD, E. A.; LOVERING, R. C.; LUSH, M. J.; WRIGHT, M. W.; POVEY, S. **Guidelines for Human Gene Nomenclature**. Disponível em: <<http://www.genenames.org/guidelines.html>>. Acesso em: outubro de 2009.

WAIN, H. M.; LUSH, M.; DUCLUZEAU, F.; POVEY, S. Genew: the Human Gene Nomenclature Database. **Nucleic Acids Research**, v.30, n.1, 2002.

WAIN, H. M.; LUSH, M. J.; DUCLUZEAU, F.; KHODIYAR, V. K.; POVEY, S. Genew: the Human Gene Nomenclature Database, 2004 updates. **Nucleic Acids Research**, v.32, 2004.

ZAHA, A. **Biologia molecular básica**. 3.ed. Porto Alegre: Mercado Aberto, 2001.

Anexo A - Scripts

ANEXOS

```

1 <html>
2   <head>
3     <title>.: BioNet ::.</title>
4   </head>
5   <frameset cols="*,320" frameborder="0" framespacing="0">
6     <frame name="steps" src="step01.php" noresize>
7     <frame name="fluxo" src="flow.php" noresize>
8   </frameset>
9 </html>
```

Script 7.1: index.php

```

1 <html>
2   <head>
3     <title>.: BioNet ::.</title>
4   </head>
```

```

5 <body>
6   <font face="verdana">
7     <table border="0" align="center" width="100%" height="100%">
8       <tr><td>
9         <form name="procuraDoenca" action="step02.php" method="post"
10        enctype="multipart/form-data">
11          <table border="0" align="center">
12            <tr>
13              <td colspan="2" align="center">
14                <br />
15                
16                <br /><br />
17              </td>
18            </tr>
19            <tr>
20              <td colspan="2" align="center">
21                <table bgcolor="#FFD42A" width="100%">
22                  <tr><td>
23                    <table bgcolor="#FFFFAA" width="100%">
24                      <tr><td>
25                        Enter the disease you want to search in the box below.
26                      </td></tr>
27                    </table>
28                  </td></tr>
29                </table><br />
30              </td>
31            </tr>
32          <td colspan="2" align="center">
33

```

```

34 <font color="gray" size="2"><b><i>Enter disease</i></b></font><br />
35 <input type="text" name="doenca" value="" size="60" /><br /><br />
36 </td>
37 </tr>
38 <tr>
39 <td width="50%" align="right"><input type="submit" name="search" value="Search" /></td>
40 <td width="50%" align="left"><input type="reset" name="clear" value="Clear" /></td>
41 </tr>
42 <tr>
43 <td colspan="2" align="center">
44 <br />
45 <table bgcolor="#B4B4B4" width="100%">
46 <tr><td>
47 <table border="1" width="100%">
48 <tr><td>
49 <td colspan="#" width="100%">
50 <font size="1">
51 The <b>BioNet</b> web server allows to mine and integrates
52 proteomic and genetic diseases data from the STRING and OMIM
53 search engines. Once a network has been generated, the
54 associated files can be saved for further analyses and the
55 entire mining processes is documented and saved by the web
56 server.<br />
57 The BioNet was designed by <b>Samuel Brando Oldra</b> as part
58 of the discipline named <b>Trabalho de Conclusão de Curso 2</b>
59 <b>(Bacharelado em Sistemas de Informação; Universidade de
60 Caxias do Sul, Brazil).</b>
61 </font>
62

```

```

63   </td></tr>
64   </table>
65   </td></tr>
66   </table><br />
67   </td>
68   </tr>
69   </table>
70   <form>
71   </td></tr>
72   </table>
73   </font>
74   </body>
75 </html>
```

Script 7.2: step01.php

```

1  <?php
2  date_default_timezone_set('America/Sao_Paulo');
3  require_once('nusoap/lib/nusoap.php');
4  $term = $_POST['doenca']; // Doença
5  $rel = '<b>Search term</b><br />' . $term;
6  // Conecta ao WebService
7  $wsdl = 'http://eutils.ncbi.nlm.nih.gov/entrez/eutils/soap/v2.0/eutils.wsdl';
8  $WebService = new nusoap_client($wsdl, 'wsdl');
9  $proxy = $WebService->getProxy();
10 // Testa a ocorrência de erros
11 $error = $WebService->getError();
12 echo $error ? $error : '';
13 // Realiza a busca pelo termo
14 $params01 = array('db' => 'omim', 'term' => $term);
```

```

15 $resp01 = $proxy->run_eSearch($params01);
16 ?>
17
18 <html>
19   <head>
20     <title>..: BioNet :.. </title>
21   </head>
22   <body>
23     <font face="verdana">
24       <center>
25         '"
27           <br />
28         <br /><br />
29       </center>
30
31 <?php if(count($resp01[$IdList][$Id]) > 1){ ?>
32
33   <table bgcolor="#FFD42A" width="100%">
34     <tr><td>
35       <table bgcolor="#FFFAAA" width="100%">
36         <tr><td>
37           Select the occurrence of the disease that is looking .
38         </td></tr>
39       </table>
40     </td></tr>
41   </table><br />
42   <input type="button" name="back" value="Back" onClick="history.go(-2)" /><br /><br />
43

```

```

44 <?php
45   foreach ($resp01[IdList][Id] as $idItem){
46     // Realiza a busca pelo id encontrado
47     $params02 = array('db' => 'omim', 'id' => $idItem);
48     $resp02 = $proxy->run_eSummary($params02);
49     // Apresenta a doença
50     echo '<font size="2">';
51     echo '<a href="step03.php?id=' . $resp02[DocSum][Id] . '" target="steps">';
52     echo $resp02[DocSum][Item][0][ItemContent] . '</a><br />';
53     echo $resp02[DocSum][Item][1][ItemContent] . ',<br />';
54     if ($resp02[DocSum][Item][2][ItemContent])
55       echo $resp02[DocSum][Item][2][ItemContent] . '<br />';
56     if ($resp02[DocSum][Item][3][ItemContent])
57       echo 'Gene map locus <a href="http://www.ncbi.nlm.nih.gov/Omim/getmap.cgi?1,
58 $resp02[DocSum][Id] ." target="_blank">';
59     $resp02[DocSum][Item][3][ItemContent] . ',</a><br />';
60     echo '</font><br />';
61   }
62   echo '<input type="button" name="back" value="Back" onClick="history.go(-2)" />';
63 } else {
64 ?>
65
66 <table bgcolor="#FF0000" width="100%">
67 <tr><td>
68 <table border="1" width="100%">
69 <tr><td>
70   No items found!
71 </td></tr>
72 </table>

```

```

73   </td></tr>
74   </table><br />
75   <input type="button" name="back" value="Back" onClick="history.go(-2)" />
76
77 <?php } ?>
78
79   </font>
80   </body>
81 </html>
```

Script 7.3: step02.php

```

1 <?php
2   $id = $_GET['id']; // Id da doença
3   $rel = '<b>Disease id</b><br /> . $id;
4 // Busca o relatório
5 $page = file('http://www.ncbi.nlm.nih.gov/entrez/dispmim.cgi?cmd=entry&id=' . $id);
6 foreach($page as $line)
7   $texto .= $line;
8 // Encontra proteinas
9 $er = '/\b[A-Z]{1}[A-Z0-9]{1,5}\b'; // Expressão regular
10 preg-match_all($er, $texto, $matches, PREG_SET_ORDER);
11 // Lista de palavras a serem retiradas
12 $listaRetirar = 'HTML HEAD TITLE OMIM HUMAN VIRUS TYPE TO BODY FFFFFF YAC
13 CC3300 NAME NOBR HIV AIDS TEXT DATE EDIT ANIMAL MODEL NOD SEE ALSO TARGET
14 CC3300 ONSET JOD IDDM CC3300 NIDDM OTHER II DQ0602 EIG W15 DR2 PCR NIDDM2
15 NIDDM3 NIDDM4 SNP K121Q PNDM PDMI WITH AND DEND MDW PNDI C166F R201H PNMD
16 I167L EEG F132L TNDM DCSIGN BDCA1 RNA DNA MOP020 GENE NSI BAT6 BAT7 BRCA3
17 B44 B53 LSA PBD LOH D17S74 OCCR SSCP PCR VNTR SNP I1307K BRCATA I167L PCR
18 YAC BRCA NIDDM3 ALPHA CAPAN1 QMPSF MIDD MTTL1 MTTE MELAS CELL
```

```

19 A3243G FACTOR ECG GROWTH LYNCH TFGBR2 MVCD1 FROM MH PDR VEGR NSN ESN RFLP
20 V67A C67X NSN A215 EMB0 V67A A3243G CRC LIKE BALB RBS11 U937 HL60 THE FED
21 DPED DPED LIKE IRAN MASON MODY6 CYSTS IDDM2 ILPR IDDM4 IGT IMDIAB HPC1 OR
22 HPCX1 HPCX2 HPC4 HPC5 HPC1 ALSPAC HPCX1 HPC7 HPC9 HPC10 HPC11 HPC12 HPC13
23 HPC14 HPC15 HPC14 V50 SOLUTE FAMILY ANION MEMBER BAND OF RED ASIAN SEVERE
24 FORM FEVER TRAIT ONE TIRAP DUFFY SYSTEM BETA LOCUS VIVAX PFBI PFFE1 LOCUS
25 FATTY GLOBIN DOMAIN PEP C IVS2 Q39X ACID MALP2 COS EL4 EC WHO FORM NO HPFH
26 COS SAO MSP1 ;
27 // Remove proteínas e termos repetidos
28 foreach($matches as $proteina){
29     foreach(split( , $proteinasEncontradas) as $val){
30         if($val == $proteina[0])
31             $existe = true ;
32     }
33     foreach(split( , $listaRetirar) as $val){
34         if($val == $proteina[0])
35             $existe = true ;
36     }
37     if(!$existe){
38         if(count(split( , $proteinasEncontradas)) <= 15)
39             $proteinasEncontradas .= $proteina[0] . , ;
40     }
41     $existe = false ;
42 }
43 ?>
44 <html>
45 <head>
46 <title>.: BioNet :.</title>
47

```

```

48
49      </head>
50      <body>
51          <font face="verdana">
52              <center>
53                  '"
55                  <br />
56                  <br /><br />
57                  <table bgcolor="#FFD42A" width="100%">
58                      <tr><td>
59                          <table bgcolor="#FFFFAA" width="100%">
60                              <tr><td>
61                                  Check the proteins found.<br />
62                                  You can also add or remove proteins at the list .
63                              </td></tr>
64                          </table >
65                      </td></tr>
66                  </table><br />
67                  <form name="procuraProteinas" action="step04.php" method="post"
68                      enctype="multipart/form-data">
69                      <table border="0" align="center">
70                          <tr>
71                              <td align="center">
72                                  <font color="gray" size="2"><b><i>Proteins found</i></b></font><br />
73                                  <textarea name="proteinas" cols="80" rows="4">
74                                      <?=$proteinasEncontradas?>
75                                  </textarea><br /><br />
76                          </td>

```

```

77 </tr>
78 <tr>          <td align="center">
79   <input type="submit" name="next" value="Next" />
80   <input type="reset" name="revert" value="Revert" />
81   <input type="button" name="back" value="Back" onClick="history.go(-2)" />
82 </td>
83 </tr>
84 </table>
85 </form>
86 <br />
87 <?= $text_o ?>
88 <input type="button" name="back" value="Back" onClick="history.go(-2)" />
89 </font>
90 </body>
91 </html>
92

```

Script 7.4: step03.php

```

1 <?php
2 $proteinas = $_POST['proteinas']; // Proteínas
3 $rel = '<b>Proteins studied</b><br />' . $proteinas;
4 $listaProteinas = split(' ', $proteinas);
5 ?>
6 <html>
7   <head>
8     <title>... BioNet ...</title>
9   </head>
10  <body>
11

```

```

12 <font face="verdana">
13 <center>
14   ;" />
16   <br />
17   <br /><br />
18   </center>
19   <table bgcolor="#FFD42A" width="100%">
20     <tr><td>
21       <table border="1" width="100%">
22         <tr><td>
23           Select the occurrences of proteins that you want to use.
24         </td></tr>
25       </table>
26     </td></tr>
27   </table><br />
28   <form name="mostraRede" action="step05.php" method="post"
29     enctype="multipart/form-data">
30     <input type="submit" name="next" value="Next" />
31     <input type="button" name="back" value="Back" onClick="history.go(-2)" />
32     <br /><br />
33     <font size="2">
34
35   <?php
36   // Mostra lista de proteinas
37   foreach($listaProtein as $proteina){
38     $proteina = trim($proteina);
39     if($proteina){
40       echo "<b>" . $proteina . "</b><br /><br />";

```

```

41     $proteinofile = file('http://string-db.org/api/tsv-no-header/resolve?identifier=' .
42                           '$protein . ,&species=9606');
43
44     if(is_array($proteinofile)) {
45         // Mostra lista de ocorrências da proteína
46         foreach($proteinofile as $vals) {
47             // Separa id e texto
48             $er = '(([A-Z0-9.]+)\s(.*)\s(.*)"'; // Expressão regular
49             preg_match_all($er, $vals, $matches, PREG_SET_ORDER);
50             foreach($matches as $match) {
51                 echo '<input name="proteinassSelecionadas[]" title="" . $match[1] .';
52                 echo ' type="checkbox" value="" . $match[1] . " /> ';
53                 echo $match[3] . '<br />;';
54             }
55         }
56     }
57 }
58 ?>
59 ?>
60
61 </font>
62 <input type="submit" name="next" value="Next" />
63 <input type="button" name="back" value="Back" onClick="history.go(-2)" />
64 </form>
65 </font>
66 </body>
67 </html>

```

Script 7.5: step04.php

```

1 <?php
2 $proteinassSelecionadas = $_POST['proteinassSelecionadas'] ; // Proteinas selecionadas
3 if(count($proteinassSelecionadas) > 0){
4   foreach($proteinassSelecionadas as $val)
5     $relTexto .= $val . ' ';
6   $rel = '<b>Selected proteins (Id)</b><br />' . $relTexto;
7   // Busca imagem da rede da(s) protéina(s)
8   if(count($proteinassSelecionadas) == 1)
9     $linkFile = 'http://string-db.org/api/url/network?identifier=' .
10    $proteinassSelecionadas[0];
11  elseif(count($proteinassSelecionadas) > 1){
12    for($i = 0; $i < count($proteinassSelecionadas); $i++){
13      if($i != (count($proteinassSelecionadas) - 1))
14        $listaProteinas .= $proteinassSelecionadas[$i] . ',%0A';
15      else
16        $listaProteinas .= $proteinassSelecionadas[$i];
17    }
18    $linkFile = 'http://string-db.org/api/url/networkList?identifiers=' .
19    $listaProteinas . '&limit=0';
20  }
21 } else
22 $linkFile = NULL;
23 ?>
24
25 <html>
26   <head>
27     <title>:: BioNet ::.</title>
28   </head>

```

```

29 <body>
30   <font face="verdana">
31     <center>
32       ' />
34       <br />
35       <br /><br />
36     </center>
37
38   <?php
39     if ($linkFile != NULL) {
40       // Pega o endereço da imagem
41       $urlFile = file($linkFile);
42       // Separa o id do endereço
43       $er = '#(e\_-([A-Za-z0-9]{2})[A-Za-z0-9\_-]+)\.'; // Expressão regular
44       preg_match_all($er, $urlFile[0], $matches, PREG_SET_ORDER);
45     ?>
46
47     <table bgcolor="#FFD42A" width="100%">
48       <tr><td>
49         <table border="1" width="100%">
50           <tr><td>
51             To <b>save</b> the XML file of the network, click with right-click on the link
52             <b>Download XML</b>, select <b>Save As</b> and choose a destination for the
53             file.<br />
54             Clicking on the link <b>Other files</b> will appear files available for this
55             network.<br />
56             Clicking on the link <b>Evaluate</b> you can evaluate the system.<br />
57             Clicking on the image of the network you will visualize it at STRING.

```

```

58 </td></tr>
59 </table>
60 </td></tr>
61 </table><br />
62 <center>
63 <a href="http://string-db.org/newstring-userdata/xml_summary.<?= $matches[0][1]?>.
64   xml" target="_blank" title="Download XML">
65   Download XML
66 </a>
67 :::
68 <a href="http://string-db.org/newstring-cgi/show_network_save_page.pl?taskId=
69   <?= $matches[0][1]?>" target="_blank" title="Other files">
70   Other files
71 </a>
72 :::
73 <a href="http://spreadsheets.google.com/viewform?formkey=
74   dE1HSEdvd0xJdXJzdzRFLW5aTVdBc1E6MA" target="_blank" title="Evaluate">
75   Evaluate
76 </a>
77 :::
78 <a href="#" onClick="history.go(-2)">Back</a><br /><br />
79 <a href="http://string-db.org/newstring-cgi/show_network_section.pl?taskId=
80   <?= $matches[0][1]?>" target="_blank" title="View at STRING database">
81   <img src=<?= $urlFile[0]?> />
82 </a>
83 </center>
84 <iframe name="aux" src="http://string-db.org/newstring-cgi/show_network_save_page.pl?
85   taskId=<?= $matches[0][1]?>" frameBorder="0" width="1" height="1" scrolling="no">
86 </iframe>

```

```

87 <?php } else { ?>
88
89     <table bgcolor="#FF0000" width="100%">
90         <tr><td>
91             <table bgcolor="#FF7F55" width="100%">
92                 <tr><td>
93                     No protein selected!
94
95                 </td></tr>
96             </table>
97         </td></tr>
98     </table><br />
99     <input type="button" name="back" value="Back" onClick="history.go(-2)" />
100
101 <?php } ?>
102
103 </font>
104 </body>
105 </html>

```

Script 7.6: step05.php

```

1 <?php
2 // Inicializar a sessão
3 session_start();
4 // Registrar variáveis
5 if (!session_is_registered('fluxo'))
6     session_register('fluxo');
7 else
8     $fluxo = $_SESSION['fluxo'];

```

```

9 // Texto adicionado pelo usuário
10 $textoUser = $_GET['textouser'];
11 if($textoUser)
12     $fluxo[count($fluxo)] = '<b>User comment</b><br />' . $textoUser;
13 // Texto adicionado pelo sistema
14 $textoSoft = $_GET['textosoft'];
15 if($textoSoft)
16     $fluxo[count($fluxo)] = $textoSoft;
17 // Fluxo adicionado pelo arquivo XML
18 $flowXML = $_FILES['flowXML'];
19 if($flowXML){
20     $nome = $flowXML['name'];
21     $tipo = $flowXML['type'];
22     $tamanho = $flowXML['size'];
23     $tmpNome = $flowXML['tmp_name'];
24     // Verifica se existe um arquivo XML enviado
25     if(strlen($tmpNome) > 0 and strlen($nome) > 1 and $tipo == 'text/xml'){
26         // Caminho completo de destino do arquivo XML
27         $caminho = realpath('..') . '/' . $nome;
28         move_uploaded_file($tmpNome, $caminho);
29         if(file_exists($caminho)){
30             // Abre arquivo XML e carrega fluxo
31             $xmlstr = file_get_contents($caminho);
32             $xmlDoc = new domDocument();
33             $xmlDoc->loadXML($xmlstr);
34             $xml = simplexml_import_dom($xmlDoc);
35             foreach($xml->step as $val)
36                 $fluxo[count($fluxo)] = (string) $val;
37     }

```

```

38
39 } // Remove item
40 $_removeItem = $_GET['$removeItem'];
41 if($_removeItem > -1){
42     if(count($fluxo) != 1){
43         unset($fluxo[$removeItem]);
44         array_unshift($fluxo, array_shift($fluxo));
45     }
46     else
47         unset($fluxo);
48 }
49 // Limpia fluxo
50 $limpaFluxo = $_GET['$limpaFluxo'];
51 if($limpaFluxo == 'yes')
52     unset($fluxo);
53 // Salva a sessão
54 $_SESSION['fluxo'] = $fluxo;
55 ?>
56
57 <html>
58 <head>
59     <title>:: BioNet ::</title>
60 </head>
61 <body bgcolor="#F4F4F4">
62     <font face="verdana">
63         <center>
64             <table border="1" width="100%">
65                 <tr><td>
66

```

```

67 <table border="1" style="width:100%;background-color:#FFFFAA">
68   <tr><td>
69     <font size="1">
70       To <b>erase</b> the current data mining click on <b>Clean flow</b>.
71       To <b>save</b> the current data mining click on <b>Download flow</b>.
72       To <b>load</b> a previously saved data mining, click the button
73       <b>Choose...</b>, locate the corresponding XML file and click the
74       button <b>Load</b>.
75       To <b>add</b> comments about the current data mining, use the text
76       box <b>User comments</b> and click the button <b>Add</b>.
77     </font>
78   </td></tr>
79 </table><br />
80 <table border="0" height="100%">
81   <tr height="100"><td>
82     <a href="flow.php?$limpaFluxo=yes" target="fluxo" title="Clean flow">
83       <center>
84         Clean flow
85       </center>
86     </a>
87   </td>
88   <td style="text-align:center;vertical-align:middle;">
89     <a href="createXMLFlow.php" target="_blank" title="Download flow">
90       Download flow
91     </a>
92   <br /><br />
93   <font color="gray" size="2">
94     <b><i>Load flow</i></b><br />
95   </font>

```

```

96   <form name="loadFlowXML" action="flow.php" method="post"
97     enctype="multipart/form-data">
98     <input type="file" size="20" name="flowXML" /><br /><br />
99     <input type="submit" name="load" value="Load" />
100    </form>
101
102    <b><i>User comments </i></b><br />
103    <form name="mostraRede" action="flow.php" method="get"
104      enctype="multipart/form-data">
105      <textarea name="textouser" cols="30" rows="4"></textarea><br /><br />
106      <input type="submit" name="add" value="Add" />
107      <input type="reset" name="clear" value="Clear" />
108    </form>
109    </font>
110  </center>
111  </td></tr>
112  <tr height="*" valign="top"><td>
113    <font size="2">
114
115    <?php
116      // Apresenta o fluxo
117      for($i = 0; $i < count($fluxo); $i++) {
118        echo '<a href="flow.php?removeItem=' . $i . '">' .
119          '[remove]</a> , ' . $fluxo[$i] . ',<br />';
120      ?>
121
122    </font>
123  </td></tr>
124  </table>

```

```

125   </center>
126   </font>
127   </body>
128 </html>

```

Script 7.7: flow.php

```

1 <?php
2 // Inicializar a sessão
3 session_start();
4 // Registrar variáveis
5 if (!session_is_registered('fluxo'))
6 session_register('fluxo');
7 else
8     $fluxo = $_SESSION['fluxo'];
9 // Nome do arquivo
10 $filename = 'flow.xml';
11 // Criando o arquivo XML
12 $xmlDoc = new domDocument('1.0', 'utf-8');
13 $xmlDoc->formatOutput = true;
14 $flow = $xmlDoc->createElement('flow');
15 $flow = $xmlDoc->appendChild($flow);
16 for ($i = 0; $i < count($fluxo); $i++) {
17     $step = $xmlDoc->createElement('step', $fluxo[$i]);
18     $step = $flow->appendChild($step);
19 }
20 $xmlDoc->save($filename);
21 // Download arquivo XML
22 header('Content-Type: application/save');
23 header('Content-Length: ' . filesize($filename));
24

```

```
24 header('Content-Disposition: attachment; filename=' . $filename . '.xml');
25 header('Content-Type: application/xml');
26 header('Expires: 0');
27 header('Pragma: no-cache');
28 // Abrir el archivo original
29 $fp = fopen($filename, 'r');
30 fpassthru($fp);
31 fclose($fp);
32 exit;
33 ?>
```

Script 7.8: createXMLFlow.php

Anexo B - Formulário de pesquisa

Pesquisa de avaliação do sistema web BioNet

*Obrigatório

Digite seu nome completo: *

Como você avalia a resposta para a doença pesquisada? *

Atende suas necessidades.

ótimo
 bom
 regular
 ruim
 péssimo

Como você avalia as proteínas extraídas do relatório da doença? *

Estão de acordo com a doença escolhida.

ótimo
 bom
 regular
 ruim
 péssimo

Figura 7.1: Formulário de pesquisa (1/3)

<p>Como você avalia a seleção das ocorrências das proteínas da doença em humanos? *</p> <p>Atende as suas necessidades.</p> <p><input type="radio"/> ótimo <input type="radio"/> bom <input type="radio"/> regular <input type="radio"/> ruim <input checked="" type="radio"/> péssimo</p>
<p>Como você avalia a rede de interação da(s) proteína(s) apresentada como resultado da pesquisa? *</p> <p>Tem utilidade.</p> <p><input type="radio"/> ótimo <input type="radio"/> bom <input type="radio"/> regular <input type="radio"/> ruim <input checked="" type="radio"/> péssimo</p>
<p>Você conseguiu utilizar o arquivo XML da rede no Cytoscape? *</p> <p>Conseguiu importar o arquivo no Cytoscape.</p> <p><input checked="" type="radio"/> sim <input type="radio"/> não</p>

Figura 7.2: Formulário de pesquisa (2/3)

Como você avalia as informações disponibilizadas no site? *

As informações foram suficientes para poder utilizá-lo.

ótimo
 bom
 regular
 ruim
 péssimo

Como você avalia a navegabilidade do site? *

O sistema é fácil de ser utilizado.

ótimo
 bom
 regular
 ruim
 péssimo

Você teria alguma sugestão de melhoria ou comentário a fazer?

Tecnologia [Google Docs](#)

[Denunciar abuso](#) · [Termos de Serviço](#) · [Termos Adicionais](#)

Figura 7.3: Formulário de pesquisa (3/3)